

Proyecto de Sistemas Informáticos,
Facultad de Informática,
Universidad Complutense de Madrid:

Asignación de Docencia SIC

Autores:

Beatriz Alonso Medina
Juan Luis Martínez Cesteros
Eduardo Renedo Gamero

Profesor director:

Manuel Núñez García

Curso:

2007-2008

1. Introducción

A lo largo de estos últimos años se hace cada vez más necesario el uso de las nuevas tecnologías y telecomunicaciones. Parece imposible pensar que alguien no tenga ordenador en sus casas o teléfonos móviles en sus bolsillos. Nos estamos acostumbrando cada vez más a acceder a la información sin movernos de casa, a través de Internet. ¿Quién no ha leído alguna vez el periódico sentado en el sillón de su casa con ayuda del ordenador? ¿Quién no ha hablado con alguien situado al otro lado del mundo?

Todos estos avances enfrían las relaciones sociales entre las personas, ya que si, por ejemplo, no salimos a comprar el periódico no tenemos la oportunidad de relacionarnos con nuestros semejantes. Sin embargo, no podemos negar que, a pesar de esto, las nuevas tecnologías facilitan, en gran medida, nuestros quehaceres diarios.

Hasta el día de hoy, la elección de la carga docente por parte del personal del departamento de sic de la facultad de informática, se hacía en una reunión que podía durar horas. A partir de ahora podrán, desde sus casas, elegir las asignaturas que desean impartir.

Esta aplicación permite que cada uno de los profesores se matricule, por orden de rango dentro del departamento, de las asignaturas que desee. Para ello, el administrador o encargado de gestionar la aplicación introducirá la información en el sistema cargando los archivos que la contengan. Cuando inicie el proceso, se le enviará un correo al primer profesor de la lista con su identificador y contraseña para que se pueda registrar. Una vez se haya matriculado, el sistema mandará automáticamente un mail al siguiente en la lista para que pueda iniciar su sesión. Todas aquellas asignaturas que haya elegido un profesor, al siguiente le aparecerán en rojo y no podrá cogerlas. Sin embargo, seguiremos permitiendo la posibilidad de que puedan seguir reuniéndose y en dicho acto elegir la carga docente; para ello, desde el administrador o encargado de gestionar la aplicación, podemos ir marcando por orden la elección de cada uno de los miembros del departamento.

In the last years is becoming absolutely necessary the use of the new and remote technologies in our daily life. Seems to be impossible thinking about anybody who does not have a cellular or a computer. We are getting use to access to the information at home without moving and with just a few clicks. Are someone who never had read the news at the Internet? Maybe it is the same person who had never been talking with somebody else at the other side of the World.

All of this advances are making the human relationships even colder, but we can not avoid to use them to have an easier way to complete our daily tasks.

Up to nowadays all the teachers at the SIC department of the Facultad de Informática in the Universidad Complutense de Madrid need to make a meeting to decide who was going to teach each subject, sometimes during hours. After this application everybody can do it at his home or any other place provided with access to the Internet.

This software allows to each teacher to make their subjects decisions, ordering the process by the established rank of each person. For this we need that the administrator or manager of that process fill in all the information about the teachers and the subjects, uploading the proper files with that info. Any time a teacher complete his register, the system send an e-mail to the next one, allowing him to make his register. We also let to the manager to make the process for everybody, and a new way to make it completely automatic.

2. Índice

1. Introducción	3
2. Índice	5
3. Análisis	6
3.1. Plan de fase	6
3.2. Requisitos	6
3.3. Riesgos	9
3.4. Iteraciones	12
3.5. Investigaciones tecnológicas	14
3.5.1. Descripción tecnologías empleadas	14
3.5.2. Seguridad en la Web	15
3.5.3. Apache-Tomcat	17
3.5.4. Logs para el servidor: Log4j	18
3.5.5. Eclipse SDK	20
3.5.6. Subversión	21
3.5.7. MySQL	21
3.5.8. JavaMail	21
3.5.9. Servlets	23
3.6. Diseño base de datos	24
3.7. Diseño página Web	30
3.8. Casos de uso	31
3.9. Diseño UML	42
Diagramas de Secuencia	42
3.10. Pruebas de sistema	49
4. Conclusiones	50
5. Manual del usuario	51
Identificarse en la aplicación	51
Vista general de la Web	52
Menú de opciones	53
Funcionalidades de la aplicación	54
Administración	54
Registro de asignaturas/profesores	54
Configuración de la aplicación	56
Matriculación usuarios	63
Proceso individual	63
Matriculación por el administrador	64
Matriculación automática	65
6. Ejemplos de uso	67
6.1. Ejemplo con un administrador	67
6.2. Ejemplo con un usuario	68
7. Bibliografía	69
8. Palabras clave	70
9. Autorización para la universidad	71

3. Análisis

3.1. *Plan de fase*

Al comienzo del proyecto, el reparto de trabajo a realizar durante los nueve meses que duraría la realización del proyecto, se corresponde con la realidad vivida. De tal manera que las fases quedan de la siguiente manera:

Fase	Fecha de Inicio	Fecha de Finalización
Inicio	15 de Octubre de 2007	5 de Noviembre de 2007
Elaboración	6 de Noviembre de 2007	4 de Febrero de 2008
Construcción	5 de Febrero de 2008	30 de Abril de 2008
Transición	1 de Mayo de 2008	12 de Junio de 2008

Los objetivos establecidos para cada Fase, los cuales no han cambiado a lo largo de todo el proyecto, han sido los siguientes:

- **Inicio:** Definición del proyecto, concretar la tarea de cada componente del grupo dentro del proyecto, establecimiento del alcance del proyecto y descripción detallada del mismo.
- **Elaboración:** Estudio de riesgos y definición de Casos de Uso. Construcción de prototipos para comprobar la viabilidad de los diseños previstos, revisión de los Casos de Uso en función de las decisiones de diseño que algunos de los prototipos evidenciaron como necesarias.
- **Construcción:** Implementación del diseño de la fase de Elaboración.
- **Transición:** Manual de usuario, creación del CD de instalación de la aplicación, recopilación de documentación y entrega del proyecto.

A pesar de estar claramente definido el comienzo y el final de cada una de las fases es inevitable que en algunas de las fases haya habido que mejorar parte del trabajo de alguna fase anterior, es decir, en la fase de elaboración se decidieron más funcionalidades por lo que hubo que aumentar los Casos de Uso. Al estar en continuo contacto con el cliente (departamento sic) se iban añadiendo más requisitos por lo que continuamente había que mejorar tanto la descripción del proyecto establecida al principio como, principalmente, la implementación.

3.2. *Requisitos*

Para la toma de requisitos ha sido básica la realización de un proceso de entrevistas sucesivas en la que hemos refinado todos los aspectos necesarios para nuestra aplicación. Estas entrevistas se realizaron con nuestro profesor del proyecto, Manuel Núñez, y con el Subdirector del departamento, Ismael Rodríguez Laguna, encargado de la asignación de docencia actualmente y que por tanto puede ser

considerado experto en el tema. Aunque lo deseable hubiese sido realizar esta toma de requisitos en una única sesión, eso es prácticamente imposible llevarlo a la práctica, siendo necesarias varias reuniones para aclarar puntos dudosos, comentar problemas surgidos y en las que habitualmente surgían nuevos requisitos como consecuencia de nuevas características que debe poseer la aplicación.

Estos requisitos comprenden aspectos como la normativa que siguen actualmente, los cálculos que realizan para cada uno de los casos propuestos o incluso el servidor y el sistema operativo en el que debe correr.

A continuación listamos los requisitos funcionales y no funcionales recogidos a lo largo de todo el proceso de entrevistas.

A. Funcionales

- Realizar una aplicación que realice la asignación docente de los miembros del departamento DSIC.
- Ésta será una aplicación Web en la cual los usuarios accederán mediante contraseña.
- Este proceso se puede realizar de tres maneras distintas :
 - Un proceso guiado mediante el envío de correos electrónicos a los usuarios. Éste seguirá un estricto orden de prioridad entre los miembros del departamento.
 - Un proceso en paralelo, en el que todos los profesores eligen sus preferencias a la vez. El programa, una vez finalizada las elecciones, generará una salida que cumpla las preferencias de cada profesor y cumpliendo todos requerimientos del proceso.
 - La última forma permitirá a un administrador realizar la matriculación de todos los miembros del departamento, de forma parecida a como se hace actualmente.
- El proceso separará la elección de docencia de los profesores y de los ayudantes, teniendo cada uno sus propios participantes y las asignaturas que les corresponden.
- Cada vez que se matricula una persona debe hacerlo por encima de un mínimo, que será calculado en base a la media sobre el número de asignaturas y del personal restante.
- El programa deberá evitar :
 - Matricular una asignatura ya matriculada.
 - Matricular dos asignaturas con horarios solapados.
 - Matricular más asignaturas de Sistemas informáticos de los permitidos.
 - Matricular a una persona si no cumple el mínimo de créditos establecido.

- El programa importará archivos en formato Excel que contienen la información de las asignaturas, profesores y créditos PDA.
- Añadir y/o modificar asignaturas y profesores desde la aplicación.
- Generar un archivo Excel salida que muestre la asignación de las diferentes asignaturas.
- El proceso puede incluir unos créditos de exención por méritos, denominados PDA.
- Se elegirá si se van a tener en cuenta los créditos de PDA de los profesores y/o de los ayudantes.
- Seleccionar el *epsilon* que marcará el mínimo por debajo de la media permitido para la elección, es decir, si la media es x , el mínimo del que pueden matricularse es de $x-\epsilon$.
- Elegir cuántos profesores, además de los introducidos en el sistema, se espera que trabajen el próximo año en el departamento.
- Poder configurar el máximo de asignaturas de Sistemas Informáticos que se le puede permitir matricularse a uno.
- Poder configurar para que haya una bonificación al elegir la primera asignatura de Sistemas Informáticos de una persona, como incentivo para estas asignaturas.
- Permitir fragmentar asignaturas de Doctorado hasta un máximo, también configurable.

B. No funcionales

- La aplicación Web deberá correr en un servidor que funcione bajo un sistema operativo Linux.
- El programa se apoyará en una base de datos para almacenar toda la información necesaria.
- Tanto el servidor como la base de datos deberán ser lo suficientemente rápidos para que la aplicación Web ejecute con soltura.
- La aplicación web deberá amoldarse a los archivos Excel que se usan actualmente en el departamento para este proceso de elección docente.
- El coste computacional debe ser mínimo, siendo esta una de las desventajas de usar el lenguaje de programación Java.
- La aplicación debe estar perfectamente encapsulada de forma que sea sencilla su instalación.
- La aplicación debe estar bien modulada, permitiendo futuras ampliaciones añadiendo un modulo sin alterar el resto del producto
- La aplicación debe tener una usabilidad sencilla, tanto para usuarios expertos como inexpertos, siendo ésta muy intuitiva.

3.3. Riesgos

A continuación se detallan los riesgos encontrados, y aplicables a nuestro proyecto, desde los tres puntos de vista más relevantes para ello, su identificación, tanto en el proceso inicial de búsqueda como los posteriormente añadidos. La posterior evaluación de riesgos, y la final planificación ante ellos. Hay que tener en cuenta que no hemos llevado a cabo una gestión de riesgos, por considerarla innecesaria debido al tamaño del equipo.

I. Identificación de Riesgos

Lista de riesgos y clasificación:

ID	Riesgo	Tipo de riesgo	Descripción
R1	Rotación de personal	Personal	Parte del personal con experiencia abandona el proyecto antes de que finalice
R2	Cambios de requisitos	Requerimientos	Existencia de más cambios de requerimientos de los previstos inicialmente
R3	Retrasos en la especificación	Proyecto	Retrasos en las especificaciones de interfaces esenciales
R4	Subestimación del tamaño	Requerimientos y estimación	El tamaño del requisito se ha subestimado
R5	Bajo rendimiento de la herramienta	Producto	Las herramientas CASE que ayudan al proyecto no tienen el rendimiento y las funcionalidades esperadas
R6	Personal sin conocimientos	Personal	El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad de algún requisito
R7	Bajas	Personal	Miembros del equipo no disponibles en momentos críticos
R8	Modificación de diseño por requisitos	Requisitos	Cambios de requisitos que precisan modificaciones en el diseño básico.
R9	Impacto requisitos	Requisitos	Los clientes/usuarios no comprenden el impacto de los cambios en los requerimientos
R10	Subestimación tiempo	Estimación	El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado

R11	Falta comunicación	Comunicación	El cliente no pueda participar en revisiones y en reuniones
R12	Sin presupuesto	Organizativos	Problemas financieros de la organización reducen las posibilidades de llevarlo a buen fin.
R13	Límite de tiempo	Organizativos	Finalización fuera de plazos originales
R14	Reestructuración	Organizativos y producto	La organización se reestructura y una nueva administración se responsabiliza del proyecto.

II. Evaluación de los riesgos

Probabilidad del riesgo valorada como *muy bajo* (<10%), *bajo* (10-25%), *moderado* (25-50%), *alto* (50-75%) o *muy alto* (>75%).

Efectos del riesgo valorados como *catastrófico*, *serio*, *tolerable* o *insignificante*.

Riesgos ordenados por efectos:

ID	Riesgo	Probabilidad	Efectos
R1	Rotación de personal	Muy baja	Catastrófico
R2	Cambios de requisitos	Alta	Catastrófico
R3	Retrasos en la especificación	Moderada	Serio
R4	Subestimación del tamaño	Moderada	Serio
R5	Bajo rendimiento de la herramienta CASE	Alta	Serio
R6	Personal sin conocimientos	Moderada	Tolerable
R7	Bajas	Baja	Tolerable
R8	Modificación diseño por requisitos	Alta	Serio
R9	Impacto requisitos	Baja	Insignificante
R10	Subestimación tiempo	Alta	Serio
R11	Falta comunicación	Moderada	Catastrófico
R12	Sin presupuesto	Muy baja	Tolerable
R13	Límite de tiempo	Moderada	Catastrófico
R14	Reestructuración	Muy baja	Catastrófico

III. Planificación de riesgos

Estrategias por riesgos.

ID	Riesgo	Estrategia
R1	Rotación de personal	Preparar un documento intentando reducir el trabajo a realizar o trabajar el tiempo restante con una alta sobrecarga. Intentar encontrar ayuda externa.
R2	Cambios de requisitos	Rastrear la información para valorar el impacto de los requerimientos, maximizar la información oculta en ellos.
R3	Retrasos en la especificación	Reorganizar el equipo de tal forma que se solapen el trabajo y los miembros comprendan el trabajo de los demás.
R4	Subestimación del tamaño	Alertar al usuario de las dificultades potenciales y las posibilidades de retraso.
R5	Bajo rendimiento de la herramienta CASE	
R6	Personal sin conocimientos	Intentar si es posible formar a esa persona. Si ningún miembro dispone de ese conocimiento intentar cambiar la tecnología empleada.
R7	Bajas	Reemplazar la labor de la persona en baja por otro compañero, o simplemente esperar su recuperación.
R8	Modificación diseño por requisitos	Reestructurar el diseño produciendo el menor impacto posible en la estructura de este.
R9	Impacto requisitos	Redactar un documento explicando de la forma más clara posible estos requisitos al usuario.
R10	Subestimación tiempo	Tratar de realizar un sobreesfuerzo.
R11	Falta comunicación	Intentar emplear medios alternativos en épocas de carencia de comunicación directa, como puedan ser telecomunicaciones a través de un ordenador o teléfono.
R12	Sin presupuesto	Tratar de solicitar el material que se precisa al departamento.
R13	Límite de tiempo	Teniendo en cuenta la total imposibilidad de ampliar este, habría que pensar la posibilidad de adaptarnos a septiembre como entrega.
R14	Reestructuración	Tratar de mantener nuestro proyecto con los nuevos encargados.

3.4. Iteraciones

Cada fase, a su vez, se dividió en Iteraciones, a través de las cuales controlar de manera más eficiente el desarrollo de cada una de las fases. Fueron precisamente la necesidad de añadir o alargar, o la posibilidad de eliminar una iteración, las que finalmente acabaron modificando la planificación del Plan de Fase. Se acordó por los integrantes del grupo que cada iteración durase un mes. Esto provocó que el plan de trabajo quedase definido del siguiente modo:

Fase	Número de Iteraciones
Inicio	1
Elaboración	3
Construcción	3
Transición	2

Iteración 1:

15 de Octubre de 2007 al 5 de Noviembre de 2007.

Esta iteración abarcó por completo la fase de Inicio del proyecto. En ella, tal y como se recoge en la lista de objetivos para dicha fase, se empezaron a recoger ideas de las entrevistas con el cliente acerca de qué podría conformar el proyecto.

Con el objetivo de optimizar esa delimitación del sistema, se realizó, previo análisis de los conocimientos de cada uno de los miembros del grupo en las distintas áreas que conformarían el proyecto, la recopilación de documentación tratando de analizar los siguientes aspectos:

- El sistema de interfaz gráfico que resultaría más provechoso para nuestros intereses.
- El Sistema de Gestión de Bases de Datos más conveniente.

Una vez se completó la investigación preliminar (que en algunos casos se prolongó durante una o dos iteraciones más), se elaboraron documentos de resumen en los que se explicaba lo descubierto en las pesquisas efectuadas.

Iteración 2:

6 de Noviembre de 2007 al 5 de Diciembre de 2007

En esta primera iteración dentro de la fase de elaboración se continuó concretando los detalles del proyecto. Una vez establecido el boceto general que especifica con detalle en qué consistiría el proyecto, se pasó a concretar esa idea, empezando por la definición del alcance del proyecto, así como de la especificación más detallada de todo aquello con lo que nuestro sistema debería contar.

Una vez ese trabajo estuvo completo, se pasó a diseñar los Casos de Uso que definen la funcionalidad del proyecto. Una vez terminados estos nos pusimos a estimar los riesgos con los que la elaboración del proyecto podría toparse.

Iteración 3:

6 de Diciembre de 2007 al 8 de Enero de 2008

En esta iteración se comienza el diseño. Tras la investigación de la fase anterior y ver qué tecnologías nos iban a ser necesarias, nos pusimos a realizar prototipos con la funcionalidad que en un futuro necesitaríamos. Estos prototipos se emplearon principalmente para estudiar cómo trabaja java con archivos Excel, para empezar a definir las estructuras de base de datos que necesitaríamos. Además fue una primera toma de contacto con servlets y jsp.

Iteración 4:

9 de Enero de 2008 al 4 de Febrero de 2008

En esta última iteración dentro de la fase de elaboración se sigue con la realización de prototipos. Tras las nuevas necesidades que iban surgiendo tuvimos que ampliar los Casos de Uso y revisar los riesgos que podrían surgir.

Iteración 5:

5 de Febrero 2008 al 3 de Marzo de 2008

Comienza la fase de construcción y con ella se empieza la implementación del código. Durante este período se escribe el código relativo a los primeros casos de uso, a los más intuitivos y básicos, es decir, se comienza a leer del archivo Excel real con el que trabajan en el departamento, se escriben los métodos necesarios para poder crear, conectar, desconectar y hacer las primeras consultas con la base de datos; y comienzan a hacerse la primera versión del los servlets y sus respectivas paginas que permitan el logeo y alguna otra gestión.

Iteración 6:

4 de Marzo de 2008 al 1 de Abril de 2008

Durante estas semanas se realiza gran parte de la implementación del proyecto, empieza a funcionar cumpliendo la mayoría de los requisitos establecidos por el cliente. Además, dadas las continuas reuniones con el departamento se incrementa la funcionalidad teniendo que añadir nuevos casos de uso y teniendo, por tanto, que revisar los posibles riesgos que pueden surgir.

Iteración 7:

2 de Abril de 2008 al 30 de Abril de 2008

En esta última etapa se mejora la funcionalidad y la eficiencia del proyecto en general. Además, se le añaden modificaciones tales como que muchos de los parámetros, con los que el departamento hace los cálculos para el reparto, sean configurables. Se realizan pruebas con casos reales depurando fallos y corrigiendo posibles errores en tiempo de ejecución.

Iteración 8:

1 de Mayo de 2008 al 2 de Junio de 2008

Comienza la fase de transición. Se empieza a escribir la documentación relativa a todo lo realizado en el proyecto. Tras la reunión anual del departamento en la que eligen la carga docente, se toman los datos y, a continuación, se pasa a efectuar una simulación de un caso real para ver si soporta con fidelidad los requisitos solicitados. Se concluyen los últimos retoques y se da por concluido lo relacionado con la implementación de código.

Iteración 9:

2 de Junio de 2008 al 20 de Junio de 2008

Cerrada la fase de Construcción se procede a realizar toda la documentación necesaria, incluido el manual de usuario. A pesar de ello, no se abandona el código y se continúan realizando pruebas para contemplar todos los posibles problemas que puedan surgir.

3.5. Investigaciones tecnológicas

3.5.1. Descripción tecnologías empleadas

Para la realización de la aplicación Web hemos decidido usar JSP debido a los conocimientos previos de Java que ya poseemos. Ya que java es un lenguaje multiplataforma nuestro proyecto también lo será, y podrá ser usado tanto en plataformas Linux como Windows.

La interacción de la parte Web con resto del código Java se realizará mediante Servlets que servirán de nexo entre ambas partes. Los servlets están también escritos en Java, con los beneficios que ello conlleva.

Mediante JSP y el uso de Servlets tenemos todo lo poder para realizar la aplicación Web requerida.

Para el estilo de la aplicación Web hemos decidido realizarla mediante CSS, inspirando el estilo de nuestra aplicación en el de la página de la facultad, manteniendo así su imagen corporativa.

Para acciones de poca importancia y a nivel de cliente, usamos código JavaScript ya que así quitamos peso de ejecución sobre el servidor.

3.5.2. Seguridad en la Web

Resolviendo la manera de identificarse y desconectarse de forma adecuada y elegante

La mayoría de las páginas Web no dedican grandes apartados a tratar de forma segura la información personal y/o confidencial que a través de ella se envía, como números de cuentas o tarjetas de crédito (que es el que más nos afecta). Otro ejemplo es el uso de ordenadores compartidos, donde un usuario que accede posteriormente puede ver datos del que estuvo anteriormente sin mayor complicación, lo cual, aunque parezca mentira, ocurre en la mayoría de las páginas de bancos españoles que hemos probado.

Ejemplos similares ocurren al acceder desde sitios públicos como bibliotecas, o la propia facultad. Y es en este tipo de lugares donde se debe prestar mayor atención a la protección de datos, asumiendo además que el usuario no tiene por qué tener ningún tipo de entrenamiento o conocimientos especiales sobre seguridad para que se le pueda garantizar que lo es.

Luego una aplicación Web perfecta, en cuanto a la protección de datos, debería comportarse de la siguiente manera: el usuario escribe la dirección de la página, y lo primero que se le muestra es una página para identificarse con sus credenciales, y asumiendo que estos son correctos la aplicación Web le permitiría acceder a todas las funcionalidades propias de su nivel de autorización. Cuando el usuario termina debe poder apretar un botón para proceder a su desconexión del sistema, y a partir de ahí lo único que se debería mostrar a cualquier usuario en cualquier condición sería la página de identificación anterior. Hay además que asumir que el usuario puede olvidar presionar este botón, o simplemente cerrar el explorador Web. Además un usuario posterior con malas intenciones podría intentar ver el contenido de la página, que ya se encuentra en la caché del ordenador presionando el botón de 'Atrás' en el navegador

Una vez vista como debería comportarse la aplicación desde el lado del usuario, veamos como se pueden llevar a cabo esos distintos requisitos desde el lado del desarrollador. En primer lugar, hay dos métodos ampliamente diferenciados para llevar a cabo todos esos puntos, la seguridad del contenedor o la personalizada. La primera hace referencia a la que ofrece un servidor de aplicaciones Web (en nuestro caso Tomcat 5.5), y la segunda a la que el desarrollador puede crear con cualquier otra herramienta. La primera ofrece rapidez y robustez, pero carece de la flexibilidad de la segunda, que es la que hemos decidido desarrollar.

La práctica más común, y la que desarrollamos nosotros, es la de ofrecer al usuario un formulario para enviar sus credenciales, que luego es comprobado contra un LDAP(*lightweight directory access protocol*), para entornos altamente seguros, o contra una base de datos relacional. Si los credenciales son validos la acción de '*login*' almacenará en el objeto *HttpSession* cierta información que permitirá saber a la aplicación por quién está siendo utilizada.

Prevenir el acceso no autorizado a páginas JSP con seguridad.

En nuestro caso ninguna página, excepto la de identificación, es mostrada a un usuario no identificado. Así que el proceso de identificación almacena en el objeto *HttpSession* los datos del usuario, y la acción asociada al botón de desconexión hace justamente lo contrario, los elimina. De esta forma toda página lo primero que hace es comprobar que hay un usuario válido en su sesión antes de hacer nada, y en caso contrario redirecciona al usuario al '*login*'.

Hasta aquí la aplicación se comporta, según el supuesto esperado, de forma correcta a la hora de mostrar información dinámica. Pero, sin embargo, no se comporta correctamente si después del usuario haberse desconectado alguien presiona el botón 'Atrás' del navegador, para ver las páginas anteriores, lo que indica que necesitamos algo más.

Evitar que el navegador almacene las páginas

El problema es que la mayoría de navegadores modernos tienen la opción de apretar el botón 'Atrás', y simplemente muestran la información que tenían de antes, no recargan la página. Para evitar esto necesitamos decirle al navegador que no almacene nuestras páginas en la caché, lo cual es algo bueno en cuanto a seguridad, pero malo en cuanto al rendimiento a la hora de solicitar varias veces la misma página. Para esta gestión disponemos de las cabeceras HTTP 'Expires' y 'Cache-Control'. El HTTP Expires le dice al navegador cuanto tardará esa página en caducar. El HTTP Cache-Control contiene los atributos para evitar el almacenamiento en caché, y así cuando el botón atrás es pulsado se hace una nueva petición de la página a la aplicación servidora. Los descriptores necesarios para esto son:

- no-cache: fuerza a obtener nuevas copias de la página
- no-store: no almacena la página en el ordenador cliente

Así hemos conseguido que tras pulsar el botón de desconexión cualquier pulsación del botón atrás, no muestre en el navegador las páginas seguras, y redirija al usuario a la pagina de '*login*'.

Aunque no obstante seguimos teniendo problemas para llegar a nuestro ejemplo perfecto. Si el usuario presiona el botón atrás en una página que enviaba información como resultado de una acción "POST", como por ejemplo la acción de identificarse en la aplicación, los navegadores modernos mostraran un mensaje de este estilo:

- IE Explorer:
 - *Warning: Page has Expired. The page you requested was created using information you submitted in a form. This page is no longer available. As a security precaution, Internet Explorer does not automatically resubmit your information for you. To resubmit your information and view this Webpage, click the Refresh button.*
- Mozilla y FireFox:
 - *The page you are trying to view contains POSTDATA that has expired from cache. If you resend the data, any action from the form carried out (such as a search or online purchase) will be repeated. To resend the data, click OK. Otherwise, click Cancel.*

Y seleccionando esta opción reenviamos la información, lo cual conduciría, si esa información es la del usuario, a volver a estar autenticado en esa página, y evidentemente esa no es la finalidad del botón de desconexión. Por lo que necesitamos algo más.

Almacenamiento de la hora del último registro

Para resolver este problema a la hora de identificarse además de la información con los credenciales del usuario se envía un campo oculto con la hora (milisegundos pasados desde 1970) que es inicializado dinámicamente. Así, cuando se realiza la llamada a la acción de validar el usuario, lo primero que hace es comparar el valor de este campo con el último almacenado en la información del usuario en la base de datos. Y sólo cuando el valor del formulario es estrictamente mayor que el almacenado en la base de datos se considera una sesión aceptable (cuando fuera producto de una recarga, ambos valores serían iguales).

Evidentemente, para una autorización válida, el campo correspondiente de cada usuario debe ser actualizado, en la base de datos con el nuevo, cuando la autorización se efectúa.

Con estas tres medidas ya hemos conseguido un comportamiento según lo esperado en el ejemplo inicial.

3.5.3. Apache-Tomcat

Para ejecutar aplicaciones Web necesitamos un servidor, preferentemente uno de código abierto, y que soporte WTP (Web Tools Platform), así que decidimos utilizar uno de los más extendidos Apache Tomcat.

Toda la documentación, y las descargas de este producto están en la propia página [Apache Tomcat](#). Nosotros decidimos utilizar la versión 5.5.25 de este mismo, del que se puede obtener un instalador para los sistemas win32 de forma libre y gratuita.

Configuración

Se realiza mediante su instalador, siendo ésta muy simple. El objetivo es que además se pueda manejar todo esto junto con el eclipse.

3.5.4. Logs para el servidor: Log4j

Log4j es una librería open source desarrollada en Java por la «Apache Software Foundation» que permite a los desarrolladores de software elegir la salida y el nivel de granularidad de los mensajes o “logs” (logging) a tiempo de ejecución y no a tiempo de compilación como es comúnmente realizado. La configuración de salida y granularidad de los mensajes es realizada a tiempo de ejecución mediante el uso de archivos de configuración externos. Log4J ha sido implementado en otros lenguajes como: C, C++, C#, Perl, Python, Ruby y Eiffel.

Conceptos:

Niveles de prioridad de los mensajes

Por defecto Log4J tiene 6 niveles de prioridad para los mensajes (debug, info, warn, error, fatal, trace). Además existen otros dos niveles extras (all y off):

NIVELES DE PRIORIDAD (De mayor -poco detalle- a menor -mucho detalle-):

- * FATAL: se utiliza para mensajes críticos del sistema, generalmente tras guardar el mensaje el programa abortará.
- * ERROR: se utiliza en mensajes de error de la aplicación que se desea guardar, estos eventos afectan al programa pero lo dejan seguir funcionando, como por ejemplo que algún parámetro de configuración no es correcto y se carga el parámetro por defecto.
- * WARN: se utiliza para mensajes de alerta sobre eventos que se desea mantener constancia, pero que no afectan el correcto funcionamiento del programa.
- * INFO: se utiliza para mensajes similares al modo "verbose" en otras aplicaciones.
- * DEBUG: se utiliza para escribir mensajes de depuración, este log no debe estar activado cuando la aplicación se encuentre en producción.
- * TRACE: se utiliza para mostrar mensajes con un mayor nivel de detalle que debug.

EXTRAS:

- * ALL: este es el nivel más bajo posible, habilita todos los logs.
- * OFF: este es el nivel más alto posible, deshabilita todos los logs.

Appenders

En Log4J los mensajes son enviados a una (o varias) salida de destino, lo que se denomina un appender.

Existen varios appenders disponibles y configurados, aunque también podemos crear y configurar nuestros propios appenders.

Típicamente la salida de los mensajes es redirigida a un fichero de texto .log (FileAppender, RollingFileAppender), a un servidor remoto donde almacenar registros (SocketAppender), a una dirección de correo electrónico (SMTPAppender), e incluso en una base de datos (JDBCAppender).

Casi nunca es utilizado en un entorno de producción la salida a la consola (ConsoleAppender) ya que perdería gran parte de la utilidad de Log4J.

Layouts

Es el responsable de dar un formato de presentación a los mensajes.

Permite presentar el mensaje con el formato necesario para almacenarlo simplemente en un archivo de texto .log (SimpleLayout y PatternLayout), en una tabla HTML (HTMLayout), o en un archivo XML (XMLLayout).

Además podemos añadir información extra al mensaje, como la fecha en que se generó, la clase que lo generó, el nivel que posee...

Configuración

La API es totalmente configurable, ya que se realiza mediante un archivo en formato XML o en formato Java Properties (clave=valor), generalmente llamado log4j.properties.

En el siguiente ejemplo implementamos un fichero properties de configuración, y configuramos dos registros.

*CONSOLE imprimirá los mensajes en la consola por líneas (%m%n).
*LOGFILE añadirá (append) los mensajes a un fichero (aplicación.log), reservando los primeros 4 caracteres para los milisegundos en que se generó el mensaje (%-4r), entre corchetes quién generó el mensaje ([%t]), cinco espacios para la prioridad del mensaje (%-5p), la categoría del mensaje (%c) y finalmente el propio mensaje junto con un retorno de carro (%m%n).

Uso

Para utilizarlo debemos, por supuesto, importar las clases que necesitamos de Log4J en nuestro código.

A continuación debemos definir una variable estática del tipo `org.apache.log4j.Logger` con el nombre de la clase que va a escribir en el registro.

Finalmente debemos configurar el objeto `Logger`.

- * Podemos utilizar la configuración básica, invocamos el método `org.apache.log4j.BasicConfigurator.configure` que configura el registro como un `ConsoleAppender` y un `PatternLayout` también predefinido.
- * Podemos configurar la API del `Log4J` también en el mismo código.
- * Podemos leer la configuración de nuestro fichero `log4j.properties`, donde definimos el nivel mínimo que debe poseer la traza para ser almacenada en el registro, el o los `appenders` que utilizaremos, y sus correspondientes `layouts`. Por defecto, al instanciar un `Logger Log4J` busca en la raíz del `classpath` de la aplicación un fichero llamado `log4j.properties` para configurarse.

3.5.5. Eclipse SDK

El entorno de desarrollo que utilizamos para el proyecto es Eclipse SDK 3.3, conocido como Europa. Es de código abierto, y se puede obtener de forma gratuita.

Configuración

No es necesario ninguna instalación del producto, simplemente su descompresión en cualquier directorio, pero sí necesitará una pequeña labor de configuración e instalación de plugins.

Plugins y programas necesarios

- Necesitamos del proyecto Web Tools Platform (WTP), que puede ser descargado bien directamente de <http://download.eclipse.org/webtools/downloads/> o instalado mediante el asistente de Eclipse (Europa discover site).
 - Si se encuentra alguna dificultad en el proceso, en la pagina de descargas del proyecto WTP, en la última release disponible se incluye un 'all in one' un paquete relativamente pesado que ya contiene un Eclipse con todos los plugins necesarios
 - Información más detallada en la documentacion de WTP: [doc-wtp](#)
- Servidor de aplicaciones web Apache Tomcat. No es propiamente un plugin para Eclipse, pero haremos que pueda funcionar como tal. Documentación en [doc-apache-tomcat](#).
- Subclipse. Plugin de Tigris para desarrollar en equipo, mediante SVN. Descargable gratuitamente a través de [tigris](#)
- Java JDK, necesario, en cualquier versión superior o igual a la 1.4.2. Se ha usado el 'Java SE Development Kit 6 Update 4' disponible gratuitamente desde [aquí](#)

Configuración

Después de lanzar Eclipse por primera vez, se necesita realizar una pequeña configuración. Si no está ya definido se necesita configurar las preferencias de Java para apuntar a su instalación del JDK.

3.5.6. Subversión

Para el desarrollo concurrente utilizamos la tecnología Subversión, y las herramientas que para ello polarion, y tigris ofrecen de forma gratuita, y que básicamente son dos:

- Subversive: Plugin para eclipse, para el trabajo en equipo a través de SVN: [<http://www.eclipse.org/subversive/>]
- TortoiseSVN: visor de carpetas SVN que se integra con el explorador de windows: [<http://tortoisesvn.tigris.org/>]

Como servidor SVN utilizamos los de Google code: [<https://lentilgroup.googlecode.com/svn/>]

3.5.7. MySQL

La base de datos que utilizamos está basada en los estándares de MySQL, en su versión 5.x, y la manejamos mediante el propio código implementado en Eclipse, para lo cual es necesario utilizar un conector para java.

Se necesitan pues un servidor MySQL, obtenible desde <http://dev.mysql.com> (MySQL 5.0 Community Server), y lo recomendable es utilizar el instalador para Windows, ya que tiene un asistente de configuración.

El conector también lo podemos obtener de la página de MySQL, y será necesario agregar la librería .jar que descarguemos (versión 5.1 a la fecha) a nuestro proyecto.

3.5.8. JavaMail

JavaMail es una expansión de Java que facilita el envío y recepción de e-mail desde código java. Usando la información de una cuenta mail cualquiera se puede realizar el envío y la recepción de mails desde una aplicación java siempre y cuando el servidor smtp de dicha cuenta lo permita.

Para su uso se requiere el uso de una librería propia del programa y la de JavaBeans Activation Framework.

Para el envío de mensaje usado en la práctica, requiere la información precisa de la cuenta a través de la cual se enviarán los mensajes. Una cuenta de gmail, por ejemplo necesitara de:

- Usuario y contraseña.
- Servidor smtp.
- protocolos de seguridad TLS si está disponible.
- Puerto 587, que no es el por defecto de smtp.
- Requiere autenticación.

El uso de la aplicación es bien sencillo, pero requiere el conocimiento de los parámetros que requiere tu cuenta de correo. Un ejemplo de su uso sería este:

```
// Usamos la clase properties, que asocia a un string un valor para ese string.  
Properties props = new Properties();
```

```
// Nombre del host del correo  
props.setProperty("mail.smtp.host", host);
```

```
// TLS si está disponible  
props.setProperty("mail.smtp.starttls.enable", ""+tls);
```

```
// Puerto para envío de correos  
props.setProperty("mail.smtp.port", ""+port);
```

```
// Nombre del usuario  
props.setProperty("mail.smtp.user", from);
```

```
// Si requiere o no usuario y password para conectarse.  
props.setProperty("mail.smtp.auth", ""+auth);
```

```
// Obtener la session  
Session session = Session.getDefaultInstance(props);  
session.setDebug(true);
```

```
// Definimos el mensaje  
MimeMessage message = new MimeMessage(session);
```

```
// añadimos el remitente  
message.setFrom(new InternetAddress(from,nom));
```

```
// añadimos los destinatarios  
for (int i=0;i< to.size();i++){  
    message.addRecipient(Message.RecipientType.TO,new  
        InternetAddress(to.get(i)));  
}
```

```
// ponemos el asunto
message.setSubject(Subject);

// y el texto
message.setText(Text);

// Y lo enviamos
message.saveChanges();
Transport transport = session.getTransport("smtp");
transport.connect(from, password);
transport.sendMessage(message, message.getAllRecipients());
transport.close();
```

3.5.9. Servlets

Los servlets son objetos que corren dentro del contexto de un contenedor de servlets (en nuestro caso Tomcat, de Apache) y extienden su funcionalidad. También podrían correr dentro de un servidor de aplicaciones (ej: OC4J Oracle) que además de contenedor para servlet tendrá contenedor para objetos más avanzados como son los EJB (Tomcat sólo es un contenedor de servlets).

La palabra servlet deriva de otra anterior, applet, que se refería a pequeños programas escritos en Java que se ejecutan en el contexto de un navegador Web. Por contraposición, un servlet es un programa que se ejecuta en un servidor.

El uso más común de los servlets es generar páginas Web de forma dinámica a partir de los parámetros de la petición que envíe el navegador Web. Un ejemplo muy claro en nuestra Web, es comprobar los parámetros de registro de los usuarios, haciendo que luego la Web responda en función a los posibles errores que esto puede generar.

Aspectos técnicos

Un servlet es un objeto que se ejecuta en un servidor o contenedor JEE, fue especialmente diseñado para ofrecer contenido dinámico desde un servidor Web, generalmente es HTML. Otras opciones que nos permiten generar contenido dinámico son con los lenguajes ASP, PHP, JSP (un caso especial de servlet) y Python.

Los servlets forman parte de JEE (Java Enterprise Edition), que es una ampliación de JSE (Java Standard Edition), por lo que necesitaremos para utilizarlos un 'jdk'.

Un servlet es un objeto Java que implementa la interfaz `javax.servlet.Servlet` o hereda alguna de las clases más convenientes para un protocolo específico (ej: `javax.servlet.HttpServlet`). Al implementar esta interfaz el servlet es capaz de interpretar los objetos de tipo `HttpServletRequest` y `HttpServletResponse` quienes contienen la información de la página que invocó al servlet.

Entre el servidor de aplicaciones (o web content) y el servlet existe un contrato que determina cómo han de interactuar. La especificación de éste se encuentra en los JSR (Java Specification Requests) del JCP (Java Community Process).

La especificación en la que están basados nuestros servlets es la 2.2, existen algunas más modernas, pero con igual funcionalidad, para lo que la empleamos.

3.6. *Diseño base de datos*

Especificación de requisitos

- Deseamos crear una base de datos para el departamento de Sistemas Informáticos y Computación con las asignaturas que imparte y su profesorado.
- Por cada profesor del departamento, identificado por un código único, deseamos almacenar su nombre, categoría (prioridad) así como un mail de contacto y una contraseña para el acceso a la aplicación Web. También será necesario especificar si es un profesor o un ayudante, ya que sus asignaturas son diferentes.
- Por cada asignatura, identificada por un código único, almacenaremos el centro en la que se imparte (Informática, matemáticas), titulación a la que pertenece (II, IGest, ISist), curso, nombre, el grupo y el cuatrimestre. También incluirá si corresponde a un profesor o un ayudante y los créditos referidos sólo a estos parámetros, y no los créditos del total de la asignatura. Por última, almacenaremos la información de horario, edificio y aula del mismo.
- El administrador de la base de datos identificado por un código almacenará al menos una contraseña.
- Queremos almacenar para cada profesor las asignaturas elegidas por él y al contrario, para cada asignatura qué profesor la imparte.

Modelo Entidad-Relación

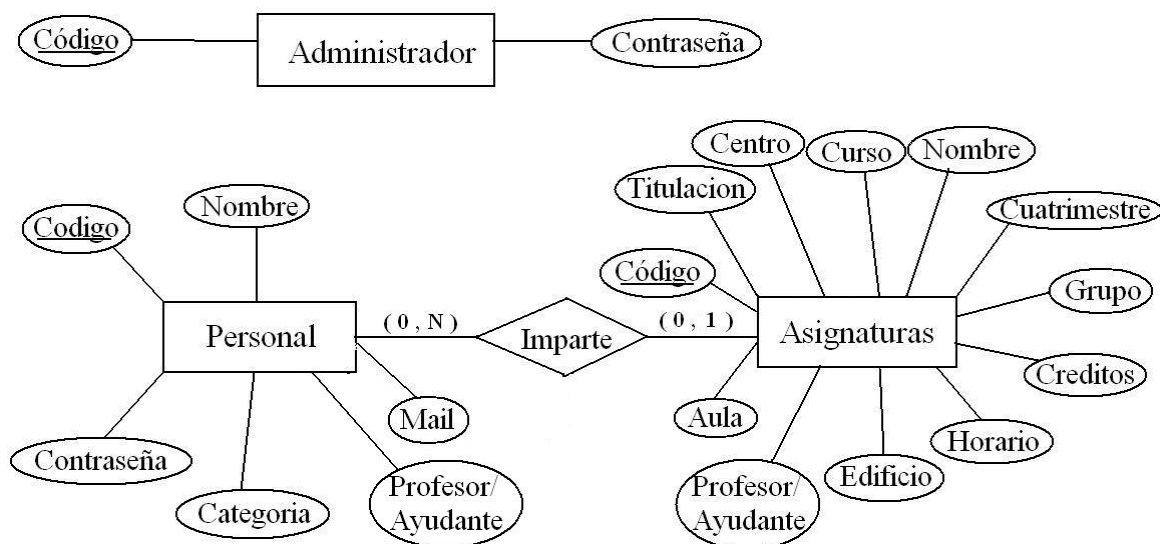
Tabla entidades

Entidad	Descripción	Atributos	Clave
Personal	Profesorado del departamento	Código, nombre, categoría, mail, contraseña, profesor/Ayudante	Código
Asignatura	Asignaturas impartidas por el departamento	Código, centro, titulación, curso, nombre, grupo, cuatrimestre, profesor/ayudante, créditos, horario, edificio, aula	Código
Administrador	Administrador de la aplicación Web	Código y contraseña	Código

Tabla relaciones

Relación	Descripción	Entidades involucradas	Atributos
Imparte	Asocia un profesor a una asignatura	Personal y Asignaturas	

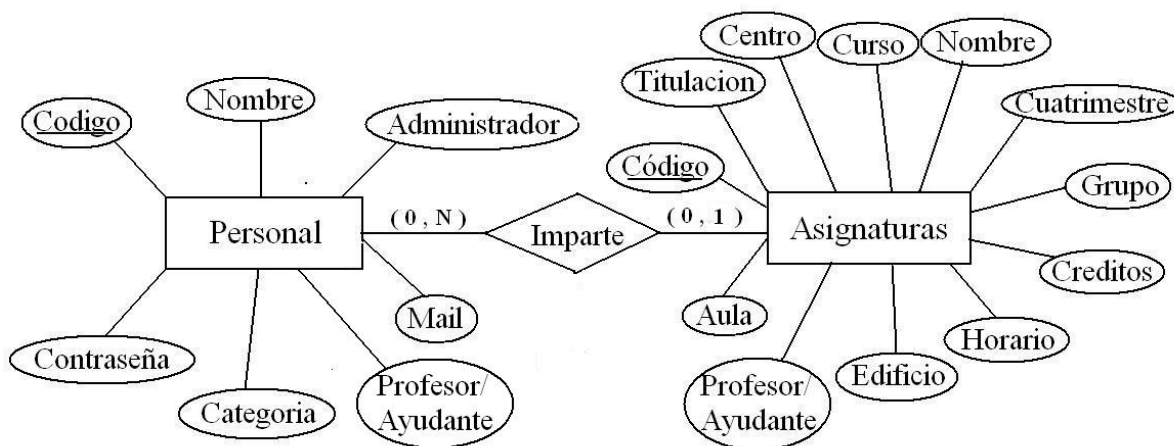
Diagrama Entidad-Relación inicial



Simplificaciones

Considerando que el administrador de la aplicación será normalmente un miembro del departamento, podemos integrar la entidad administrador dentro de personal, añadiendo a esta un nuevo atributo, administrador, de tipo booleano.

Diagrama Entidad-Relación final



Modelo Relacional

El modelo relacional se obtiene de manera directa, y de él sacaremos las tablas a realizar en nuestra base de datos. Como en la relación Imparte el elemento Asignatura tiene una relación de 0-1, podemos ahorrarnos la tabla de Imparte e incluir la clave de Personal dentro de la Tabla Asignatura, pudiendo ser este atributo Nulable.

La clave primaria se pone en negritas, y las claves ajenas dependientes de los valores de otra tabla se marcan con subrayado.

Quedaría así:

Personal (**Código**, Nombre, Administrador, Contraseña, Categoría, Mail)

Asignaturas (**Código**, Centro, Titulación, Curso, Nombre, Cuatrimestre, Grupo, Profesor/Ayudante, Créditos, Horario, Edificio, Aula, CódigoPersonal)

Normalización

El objetivo de la Normalización es el evitar el mayor número de redundancias de datos posibles en una base de datos, por lo que en una aplicación de bajo nivel como esta tiene poca importancia. Ya que en el peor de los casos la base de datos contendrá poco más de un centenar de asignaturas y unas decenas de profesores, en nuestra aplicación podemos permitirnos un elevado ratio de repetición de datos en favor de una mayor claridad de la estructura de los datos almacenados en ella.

Aún así vamos a estudiar en que forma Normal se encuentra la base de datos.

Podemos ver fácilmente que nuestra tabla cumple la primera forma Normal, ya que todos sus atributos son atómicos y toda tabla posee una clave primaria. A veces se considera que no puede haber atributos nulos en las formas Normales, pero en nuestra aplicación es algo tan habitual que no lo consideramos.

La segunda forma Normal exige que todo atributo no perteneciente a la clave dependa totalmente de ella, lo cual es cierto en nuestra aplicación, ya que la clave primaria es un atributo únicamente, Código, y sin ella podría haber filas duplicadas, como en el caso de las asignaturas de Sistemas Informáticos.

La 3FN exige que ningún atributo no perteneciente a la clave dependa transitivamente de la clave. En la tabla Personal si suponemos que puede haber dos personas con el mismo nombre entonces se cumple. En la tabla Asignatura es difícil de comprobar esto debido al elevado número de atributos que posee y por tanto a la cantidad de relaciones que podría haber entre ellos. Pero parece que siempre se puede encontrar un ejemplo en el que para un atributo dado los otros pueden cambiar, por lo que parece que ningún atributo dependa de otro, y por tanto no parece posible transitividad. Esto se refuerza con la idea de que cada atributo puede ser modificado de manera independiente de los demás, por lo que parece que nuestra tabla no posee relaciones internas. Por esta misma razón nuestra base de datos también cumple la Forma Normal de Boyce-Codd.

La 4NF se asegura de que los hechos multivalores independientes estén correcta y eficientemente representados en el diseño de base de datos. En este caso nuestra base de datos no parece cumplir este requisito, ya que es habitual poseer asignaturas casi idénticas con cambios en un único valor incluso iguales, por lo que no se cumple esta forma normal ni las posteriores.

Ampliaciones

Una vez investigadas todas las tecnologías a usar y debido a algunos requisitos funcionales necesarios en la aplicación Web, ha sido necesario introducir una serie de parámetros adicionales a nuestra definición inicial de la base de datos.

En primer lugar, para el envío de mail mediante Jmail es necesario tener almacenados una serie de parámetros de configuración para que esta aplicación pueda realizar el envío de correos. Estos parámetros serían:

- Cuenta de correo electrónico que enviará los correos electrónicos.
- Contraseña de dicha cuenta.
- Servidor smtp de la cuenta de correo.
- Puerto usado por este servidor para el envío de correos electrónicos, normalmente el 25, pero cambia en algunos casos.
- Si es requerida o no la autenticación de la cuenta.
- Si es requerido o no el uso de TLS, un protocolo de comunicación segura.

Estos parámetros se incluirían dentro una entidad propia, CorreoMail.

Nuestra aplicación requería ser configurable para ciertos parámetros típicos, que pueden cambiar de un año para otro. Para poder llevar un control sobre estas configuraciones, se decidió crear una nueva tabla, INFOADMIN, donde almacenar la configuración elegida para una determinada elección de docencia. Esta tabla incluye los siguientes parámetros:

- estadoproceso: para poder saber en qué estado se encuentra la elección de docencia. Estos estados serían: antes de haber comenzado, estando ya en proceso, y ya terminado.
- pagina web: contiene la dirección Web donde se encuentra alojada la aplicación, para poder comunicársela a los participantes y que puedan acceder a ella.
- tipo de aplicación: Hay dos formas de usar la aplicación, de una manera lineal, como se viene haciendo hasta ahora, o de una manera en paralelo.
- epsilon: valor necesario para el cálculo de valor de créditos mínimo para matricularse.
- primersisinf: incentivo a la primera asignatura de sistemas informáticos que se ha matriculado.
- numerosisinf: cantidad máxima de asignaturas de sistemas informáticos que se le permite matricular a un profesor.
- maxcredfrag: máximo número de crédito que se le permite coger a un profesor cuando fragmenta una asignatura.
- doctoradfrag: para considerar a todas las asignaturas de doctorado como fragmentables o no.
- pdaprofesores: para marcar si se tendrá en cuenta la exención por PDA en la asignación docente de los profesores.
- pdaayudantes: para marcar si se tendrá en cuenta la exención por PDA en la asignación docente de los ayudantes.
- estimaprof: estimación para los profesores contratados para el año que viene. Influye en las cuentas.
- estimaayud: estimación para los ayudantes contratados para el año que viene. Influye en las cuentas.
- estadoimportación: estado que se encuentra la importación de los distintos archivos. Necesario para impedir la lectura anticipada de archivos que dependen de otros.
- CorreccionCuentas: Al cálculo de los créditos totales para contabilizar se suelen hacer correcciones debido a asignaturas que no las dará el departamento, y que se pasaran por ejemplo al departamento GSME.

Por motivos de seguridad en el login de la aplicación Web, necesitamos almacenar la hora del último login para al manejarla evitar la recuperación de una sesión

anterior por parte de otra persona. Este parámetro lo incluiremos dentro de los parámetros de la tabla personal.

Para llevar un control de que profesor esta matriculado ya y quien no, necesitamos un nuevo parámetro booleano para marcar esta situación en personal.

En la aplicación, un administrador puede acceder a una información que los demás no pueden ver. Por tanto, necesitamos poder marcar quien es administrador. También en la tabla personal.

Un requisito funcional de la aplicación es la contabilización de la exención de créditos por PDA, que pueden ser usadas o no en el proceso de elección de docencia. Por tanto este parámetro también ha de ser introducido, y como es relativo a cada profesor, lo incluimos en la tabla personal.

Las asignaturas de doctorado tienen una propiedad interesante, que es la de poder ser fragmentadas y asignadas a varios profesores. Esta propiedad, que hemos definido como FRAGMENTABLE se resuelve con un booleano para marcar las asignaturas que poseen esta propiedad.

Dado que las asignaturas pueden ser de carácter teórico, practico, o tener ambas partes, como sucede en los laboratorios. Decidimos incluir un parámetro TIPO pensando sobre todo en este último caso, para poder tratar como una sólo asignatura ambas partes, la teórica y práctica de un laboratorio.

El modelo Relacional final queda así:

Personal (Código, Nombre, Administrador, Contraseña, Categoría, Mail, Profesor/Ayudante, Administrador, Matriculado, PDA, LastLogon)

Asignaturas (Código, Centro, Titulación, Curso, Nombre, Cuatrimestre, Grupo, Profesor/Ayudante, Créditos, Horario, Edificio, Aula, CodigoPersonal, Fragmentable, Tipo)

CorreoEnvio (Nombre, Cuenta, Contraseña, Servidor, Puerto, ServidorSMTP, TLS, Autenticación)

InfoAdmin (paginaWeb, estadoProceso, epsilon, tipoAplicación, primerSisInf , NumSisInf, doctoradoFrag, maxCredFrag, PDAProfesores, PDAAyudantes, EstimaProf, EstimaAyud, EstadoImportacion, correccionCuentas)

3.7. *Diseño página Web*

Idea inicial

La aplicación consistiría en una aplicación web con dos formas de acceder, una para el administrador del sistema y otra para los usuarios que van a seleccionar sus asignaturas.

Administrador

El administrador se encargaría de preparar la aplicación para su correcto funcionamiento. Para esto introducirá los datos necesarios, referentes a los profesores que van a participar en el proceso y los distintos puestos o asignaturas que van a poder solicitar. Para facilitar esta labor la introducción de datos se hará a través de un archivo Excel con el formato adecuado.

La aplicación contará también con una cuenta de correo electrónico para poder mandar la información necesaria a los participantes. Es, por tanto, labor del administrador preparar la configuración de esta cuenta de correo.

Por último, el administrador, con todo listo, procedería a comenzar el proceso de matrícula, configurando las distintas características que quisiera aplicar al proceso de selección.

Usuarios

Cada participante recibiría en su turno un mail con su nombre identificador y una contraseña para acceder a la aplicación. Una vez dentro, tendrá una lista de todas las asignaturas o puesto todavía vacantes que puede solicitar, con un mínimo de créditos necesario para poder llevar a cabo la matrícula.

La página contendrá búsquedas, información de cada asignatura, y horarios como facilidades para el usuario.

Una vez terminada la elección se almacenará esta y se procederá al envío de un mail al siguiente participante.

Final

Al acabar el proceso de matrícula, se procederá a sacar la solución obtenida mediante un archivo Excel, con todos los cálculos que sean necesarios.

Hojas de estilo para la Web

Básicamente, nuestra página mostrará apartados similares a los de la página de la facultad.

Una cabecera:

En todo momento contendrá un menú para seleccionar distintas formas de ver la web, incluyendo algunas para personas con menor accesibilidad (alto contraste o mayor tamaño de letra, y paginas con solo texto), Emblemas corporativos, que pueden ser los del departamento por ejemplo, y otro menú con las acciones correspondientes al tipo de usuario que está accediendo (según si no está registrado, es un usuario común, o es el administrador).

Un bloque con opciones a la izquierda:

Muestra información del usuario, así como sus diversas opciones en todo momento.

Un bloque central:

Muestra todo el contenido a tratar.

Un pie de página.

3.8. Casos de uso

Vamos a separar los casos de uso según el tipo de usuario. Por tanto los separaremos en dos grupos, los relativos al administrador y los relativos a un usuario normal, miembro del departamento, ya sea profesor o ayudante.

Índice de casos:

Relativos al administrador

1. Acceder a la aplicación como administrador predefinido.
2. Cargar Excel con datos.
3. Añadir personal.
4. Añadir asignaturas.
5. Editar personal.
6. Editar asignaturas.
7. Borrar personal.
8. Borrar asignaturas.
9. Configurar cuenta de correo electrónico.
10. Configurar aplicación.
11. Comenzar proceso de elección de docencia.
12. Reiniciar el proceso de elección de docencia.
13. Matricular a alguien en su nombre.

Relativos a un usuario

14. Acceder a la aplicación.
15. Buscar asignatura selectivamente.
16. Seleccionar asignatura.
17. Quitar asignatura.
18. Comprobar horario.
19. Matricularse.

RELATIVOS AL ADMINISTRADOR

[CU-01]	Acceder a la aplicación como administrador predefinido.
Objetivo en contexto	Acceder a la aplicación a través de una cuenta de administrador siempre disponible.
Entradas	Número de usuario y contraseña.
Precondiciones	
Salidas	Mensaje de error si no existe ese usuario con esa contraseña.
Poscondición si éxito	Se accede a la aplicación con privilegios de administrador.
Poscondición si fallo	Se vuelve a intentar la operación.
Actores	Usuario de Internet.
Secuencia normal	<ol style="list-style-type: none"> 1. Pantalla inicial de login. 2. Introducir identificador '1' y contraseña '1234'. 3. El sistema comprueba que existe el identificador indicado y comprueba si la contraseña es correcta. Si no es correcta ira S1. 4. La aplicación permite el acceso a la GUI de la aplicación con toda la funcionalidad del administrador.
Secuencia alternativa	S1. Mensaje de error por usuario o contraseña no válidas.

[CU-02]	Cargar Excel con datos
Objetivo en contexto	Cargar la información del archivo Excel en el sistema. Este caso de uso es válido para cualquier archivo Excel del sistema.
Entradas	Archivo Excel a cargar.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje informando de si el archivo es correcto o posee algún fallo para corregir.
Poscondición si éxito	Se introduce el contenido del archivo en la base de datos.
Poscondición si fallo	Si el archivo no es correcto, no hace nada. Si el archivo es correcto pero posee errores, el programa sugiere una solución por cada fallo.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de carga de docencia. 2. Seleccionar el archivo a cargar. 3. Pulsar Subir. 4. Se informa de la validez del archivo. Si el archivo es erróneo ir a S1. Si el archivo posee algún fallo ir a S2. 5. Se introducen los datos en la base de datos.
Secuencia alternativa	<p>S1. Se informa de que el archivo no es correcto.</p> <p>S2. Se muestran sugerencias para resolver cada fallo. Con todos los fallos corregidos se procede a introducir los datos en la base de datos.</p>

[CU-03]	Añadir personal
Objetivo en contexto	Añadir manualmente una persona en el sistema.
Entradas	Formulario con los datos necesarios para una persona.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se introduce a la persona en la base de datos.
Poscondición si fallo	No se modifica la base de datos.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de añadir personal. 2. Rellenar los parámetros deseados. 3. pulsar insertar. 4. Si el nombre de la persona esta repetido ir a S1. Si falta algún dato ir a S2. Si el formato de un dato es incorrecto ir a S3. 5. Se introducen los datos en la base de datos.
Secuencia alternativa	<p>S1. Se informa de que la persona esta ya en el sistema.</p> <p>S2. Se informa del dato que obligatorio introducir.</p> <p>S3. Se informa del formato erróneo para ese dato.</p>

[CU-04]	Añadir asignaturas
Objetivo en contexto	Añadir manualmente una asignatura en el sistema.
Entradas	Formulario con los datos necesarios para una asignatura.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se introduce la asignatura en la base de datos.
Poscondición si fallo	No se modifica la base de datos.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de añadir asignatura. 2. Rellenar los parámetros deseados. 3. pulsar insertar. 4. Si falta algún dato ir a S1. Si el formato de un dato es incorrecto ir a S2. 5. Se introducen los datos en la base de datos.
Secuencia alternativa	<p>S1. Se informa del dato que obligatorio introducir.</p> <p>S2. Se informa del formato erróneo para ese dato.</p>

[CU-05]	Edita personal
Objetivo en contexto	Editar manualmente una persona.
Entradas	Tabla con todo el personal disponible.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se modifican los datos de la persona en la base de datos.
Poscondición si fallo	No se modifica la base de datos.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de mostrar profesores. 2. Pulsar editar en el profesor elegido de entre los que aparecen en la tabla. 3. Modificar los parámetros deseados. 4. pulsar aceptar. 5. Si el nombre de la persona esta repetido ir a S1. Si falta algún dato ir a S2. Si el formato de un dato es incorrecto ir a S3. 6. Se introducen los datos en la base de datos.
Secuencia alternativa	<p>S1. Se informa de que la persona esta ya en el sistema.</p> <p>S2. Se informa del dato que obligatorio introducir.</p> <p>S3. Se informa del formato erróneo para ese dato.</p>

[CU-06]	Edita asignaturas
Objetivo en contexto	Editar manualmente una asignatura.
Entradas	Tabla con todas las asignaturas disponibles.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se modifican los datos de la asignatura en la base de datos.
Poscondición si fallo	No se modifica la base de datos.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de mostrar asignaturas. 2. Pulsar editar en la asignatura elegida de entre los que aparecen en la tabla. 3. Modificar los parámetros deseados. 4. pulsar aceptar. 5. Si falta algún dato ir a S1. Si el formato de un dato es incorrecto ir a S2. 6. Se introducen los datos en la base de datos.
Secuencia alternativa	S1. Se informa del dato que obligatorio introducir. S2. Se informa del formato erróneo para ese dato.

[CU-07]	Borrar personal
Objetivo en contexto	Borrar una persona presente en la base de datos.
Entradas	Tabla con todo el personal disponible.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se borra la persona en la base de datos.
Poscondición si fallo	
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de mostrar profesores. 2. Pulsar borrar en el profesor elegido de entre los que aparecen en la tabla. 3. Se borra de la base de datos.
Secuencia alternativa	

[CU-08]	Borrar asignaturas
Objetivo en contexto	Borrar una asignatura presente en la base de datos.
Entradas	Tabla con todas las asignaturas disponibles.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Se borra la asignatura en la base de datos.
Poscondición si fallo	
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el menú de mostrar asignaturas. 2. Pulsar borrar en la asignatura elegida de entre las que aparecen en la tabla. 3. Se borra de la base de datos.
Secuencia alternativa	

[CU-09]	Configurar cuenta de correo
Objetivo en contexto	Ajustar todos los parámetros necesarios que permiten enviar correos electrónicos desde una aplicación Java.
Entradas	Formulario con los parámetros de la cuenta de correo electrónico.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Los parámetros de correo quedan configurados en la base de datos.
Poscondición si fallo	
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el apartado de gestión de correo. 2. rellenar los campos de configuración. 3. guardar la configuración 4. Opcionalmente se puede probar el funcionamiento de la cuenta que se acaba de configurar. Si algún parámetro tiene un formato incorrecto S1
Secuencia alternativa	S1. Se informa del formato incorrecto.

[CU-10]	Configurar Aplicación
Objetivo en contexto	Ajustar todos los parámetros configurables de la aplicación antes de comenzar el proceso de elección de docencia.
Entradas	Formulario con los elementos a configurar.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de éxito o error.
Poscondición si éxito	Los parámetros quedan configurados en la base de datos.
Poscondición si fallo	
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el apartado de proceso. 2. Asegurarse que se han realizado los pasos anteriores que sugiere la página. 3. rellenar los campos de configuración. 4. guardar la configuración. Si algún parámetro tiene un formato incorrecto S1
Secuencia alternativa	S1. Se informa del formato incorrecto.

[CU-11]	Comenzar proceso de elección de docencia
Objetivo en contexto	Comienza la asignación de docencia según como se haya configurado en la aplicación.
Entradas	Parámetros previamente configurados.
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensaje de comienzo del proceso de asignación docente.
Poscondición si éxito	Se envía un correo electrónico a cada participante con la información del proceso que se va llevar a cabo, y un correo electrónico al primer participante con clave y contraseña para que pueda acceder a la aplicación Web y realice su elección.
Poscondición si fallo	El proceso no comenzaría
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el apartado de proceso. 2. Asegurarse que se han realizado los pasos anteriores que sugiere la pagina. 3. Asegurarse que la configuración es la adecuada. 4. pulsar comenzar. 5. Se envía un correo electrónico general a todos los participantes informándoles que se desarrollará el proceso. 6. Se envía un correo electrónico con la clave y la contraseña. Será únicamente al primero si el proceso es individual o a todos si el proceso es de tipo automático.
Secuencia alternativa	

[CU-12]	Reiniciar el proceso de elección de docencia
Objetivo en contexto	En caso de algún contratiempo reiniciamos la aplicación y toda a la asignación de docencia hasta ahora.
Entradas	
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Mensajes de advertencia y éxito.
Poscondición si éxito	Se desmatricula en la aplicación a todo el mundo y se desligan todas las asignaturas de sus profesores. Esto puede causar que asignaturas asignadas de primeras se reasignen, así que se recomienda recargar los archivos Excel.
Poscondición si fallo	
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. Abrir el apartado de proceso. 2. Pulsar el botón reiniciar. 3. Aparece una advertencia. 4. Si se confirma, se desmatriculan profesores y asignaturas.
Secuencia alternativa	

[CU-13]	Matricular a alguien en su nombre
Objetivo en contexto	Suplantar la identidad del siguiente participante para realizar la matrícula en su nombre.
Entradas	Formulario de asignaturas aplicable a la persona que se sustituye
Precondiciones	Usuario registrado con privilegios de administrador.
Salidas	Matriculación exitosa o mensajes de fallo diversos.
Poscondición si éxito	Si la matrícula cumple todas las condiciones impuestas por la configuración se le asignarán a la persona sustituida las asignaturas elegidas.
Poscondición si fallo	Se advierte de la regla que se esta incumpliendo para su corrección.
Actores	Administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. En el apartado de elección de asignaturas seleccionar el botón "Matricular desde cuenta de admin" 2. Se suplanta la identidad de la siguiente persona a matricularse. 3. El administrador realiza la elección de las asignaturas para la persona suplantada siguiendo el CU-16. 4. Una vez elegidas todas las asignaturas pulsar "Terminar". Si hay solapamiento de horarios ir a S1. Si se matricula menos créditos del mínimo permitido ir a S2. Si se ha partido más de una asignatura anual ir a S3. 5. El profesor suplantado matricula para sí las asignaturas elegidas. 6. Se procede a la matriculación de siguiente profesor en la lista. 7. Se puede dejar de matricular a profesores en su nombre pulsado "Dejar de Matricular"
Secuencia alternativa	<p>S1. Informar de que hay Solapamiento de Horarios en la elección de docencia realizada.</p> <p>S2. Informar de que no se ha llegado al mínimo permitido para matricular a una persona.</p> <p>S3. Informar de que hay más particiones de las permitidas.</p>

RELATIVOS A UN USUARIO

[CU-14]	Acceder a la aplicación
Objetivo en contexto	Acceder a la aplicación como un usuario normal.
Entradas	Número de usuario y contraseña.
Precondiciones	El usuario debe haber sido añadido previamente.
Salidas	Mensaje de error si no existe ese usuario con esa contraseña.
Poscondición si éxito	Se accede a la aplicación con los privilegios definidos para él.
Poscondición si fallo	Se vuelve a intentar la operación.
Actores	Usuario de Internet.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario ha debido recibir previamente un correo electrónico con su identificador, su contraseña y donde se encuentra hospedada la aplicación Web. 2. Pantalla inicial de identificación. 3. Introducir identificador y contraseña. 4. El sistema comprueba que existe el identificador indicado y comprueba si la contraseña es correcta. Si no es correcta ira S1. 5. La aplicación permite el acceso a la GUI de la aplicación con la funcionalidad permitida para los privilegios del usuario.
Secuencia alternativa	S1. Mensaje de error por usuario o contraseña no válidos.

[CU-15]	Buscar asignatura selectivamente
Objetivo en contexto	Realizar una búsqueda determinada entre todas las asignaturas incluidas en el sistema para seleccionarla.
Entradas	Formulario con desplegados
Precondiciones	El usuario esta registrado
Salidas	La tabla de asignaturas actualizada
Poscondición si éxito	La tabla de asignaturas actualizada ,apareciendo únicamente las asignaturas que cumplen todas las condiciones
Poscondición si fallo	.
Actores	Usuario registrado.
Secuencia normal	<ol style="list-style-type: none"> 1. Apartado de elección de asignaturas 2. Seleccionar los criterios de búsqueda que debe cumplir toda asignatura. Para la búsqueda por horario especificar un rango en el que deben estar todas las clases de esa asignatura. 3. La tabla se actualiza tras cada criterio nuevo
Secuencia alternativa	

[CU-16]	Seleccionar asignatura
Objetivo en contexto	Se elige una asignatura para formar parte de la asignación de docencia propia del usuario.
Entradas	Formulario con las asignaturas.
Precondiciones	El usuario esta registrado
Salidas	La tabla de asignaturas seleccionadas por el usuario actualizada
Poscondición si éxito	Se añade la asignatura a la elección docente
Poscondición si fallo	Se informa del fallo producido.
Actores	Usuario registrado.
Secuencia normal	<ol style="list-style-type: none"> 1. Apartado de elección de asignaturas 2. Si la asignatura es de doctorado se puede fragmentar. antes de seleccionarla se rellena la cantidad de créditos a coger en dicha asignatura. Si se supera el máximo permitido ir a S1. 3. Se selecciona una asignatura disponible de las mostradas en pantalla. Si al añadir la asignatura se incumple una normal del proceso ir a S2. 4. La asignatura se añade a la elección realizada por el usuario.
Secuencia alternativa	<p>S1. Informar del error explicando el máximo permitido a matricular.</p> <p>S2. Si el error es debido a superar el máximo de asignaturas de Sistemas Informáticos permitidos, se informa de éste.</p>

[CU-17]	Quitar asignatura.
Objetivo en contexto	Se elige una asignatura de la asignación de docencia propia del usuario y se elimina de ella.
Entradas	Formularios con las asignaturas elegidas por el usuario y todas las demás.
Precondiciones	El usuario esta registrado
Salidas	La tabla de asignaturas seleccionadas por el usuario actualizada
Poscondición si éxito	Se quita la asignatura de la elección docente
Poscondición si fallo	
Actores	Usuario registrado.
Secuencia normal	<ol style="list-style-type: none"> 1. Apartado de elección de asignaturas 2. Se desmarca la asignatura a quitar, ya sea en la propia tabla de la elección docente del usuario o en la tabla general.
Secuencia alternativa	

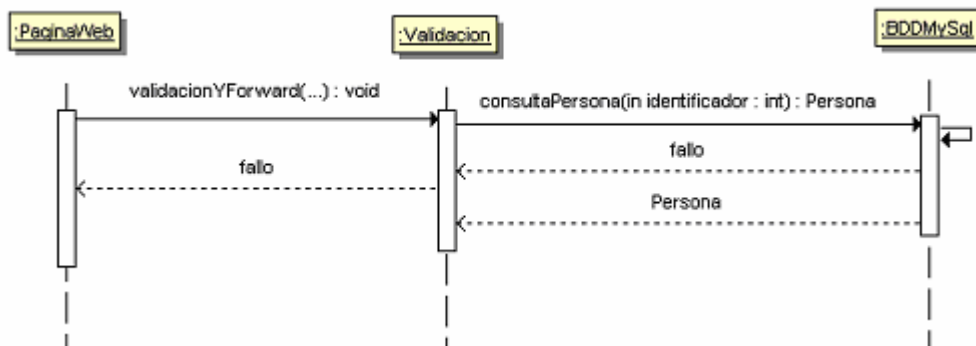
[CU-18]	Comprobar horario
Objetivo en contexto	Permite al usuario comprobar el horario que poseerá el usuario el curso académico siguiente según la elección docente que esta realizando en ese momento.
Entradas	Elección de asignaturas realizada por el usuario hasta el momento
Precondiciones	El usuario esta registrado
Salidas	Una tabla-horario con las horas de las asignaturas marcadas en el, mostrando los solapamientos si los hay.
Poscondición si éxito	Muestra la tabla-horario del cuatrimestre seleccionado
Poscondición si fallo	
Actores	Usuario registrado.
Secuencia normal	<ol style="list-style-type: none"> 1. Apartado de elección de asignaturas 2. El usuario pulsa 1º o 2º cuatrimestre según el horario que desea ver. 3. Se muestra la tabla de dicho cuatrimestre.
Secuencia alternativa	

[CU-19]	Matricularse
Objetivo en contexto	Asignarle las asignaturas elegidas al usuario que esta realizando la elección de docencia.
Entradas	Elección de asignaturas realizada por el usuario
Precondiciones	El usuario esta registrado y es la persona que posee el turno.
Salidas	Mensaje de éxito, o de fallo si se produce.
Poscondición si éxito	Si la matrícula cumple todas las condiciones impuestas por la configuración se le asignarán a la persona las asignaturas elegidas.
Poscondición si fallo	Se advierte de la regla que se esta incumpliendo para su corrección.
Actores	Usuario registrado en turno.
Secuencia normal	<ol style="list-style-type: none"> 1. Apartado de elección de asignaturas 2. Una vez elegidas todas las asignaturas pulsar "Terminar". Si hay solapamiento de horarios ir a S1. Si se matricula menos créditos del mínimo permitido ir a S2. Si se ha partido más de una asignatura anual ir a S3. 3. El profesor matricula para si las asignaturas elegidas. 4. Se envía un correo electrónico confirmando la elección docente realizada. 5. Se envía un correo electrónico al siguiente participante con la información necesaria, su clave y contraseña.
Secuencia alternativa	<p>S1. Informar de que hay Solapamiento de Horarios en la elección de docencia realizada.</p> <p>S2. Informar de que no se ha llegado al mínimo permitido para matricular a una persona.</p> <p>S3. Informar de que hay más particiones de las permitidas.</p>

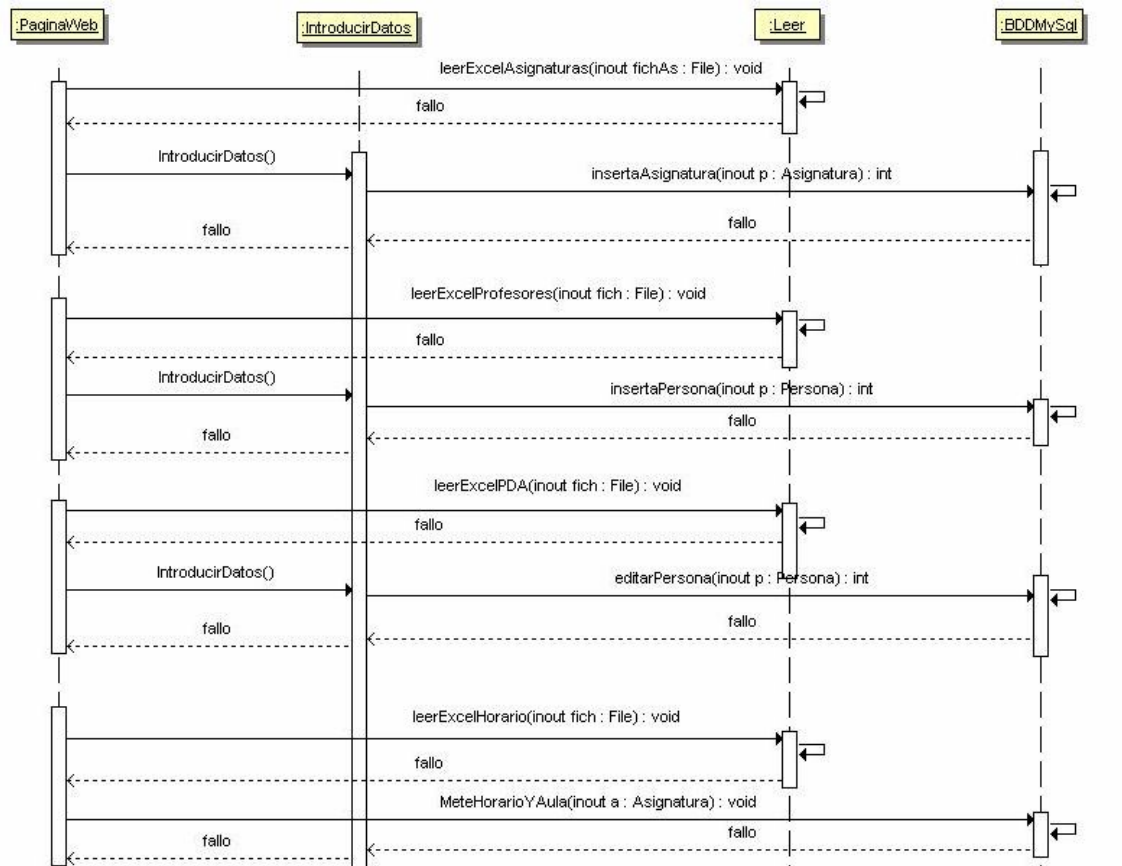
3.9. Diseño UML

Diagramas de Secuencia

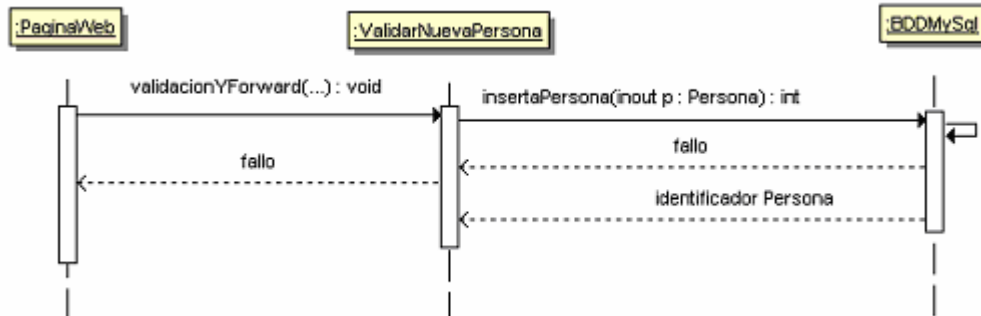
CU – 01 Acceder a la aplicación como administrador predefinido



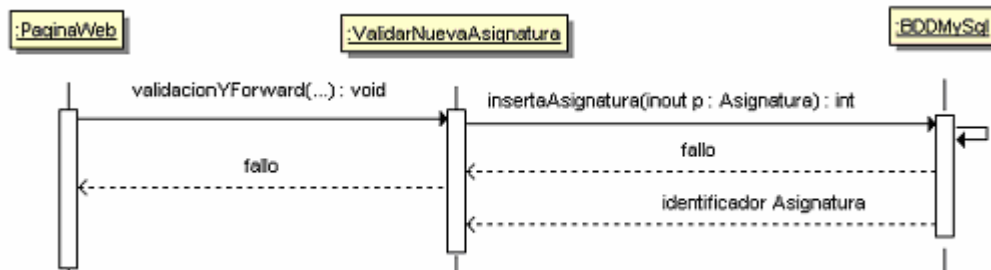
CU – 02 Cargar Excel con los datos



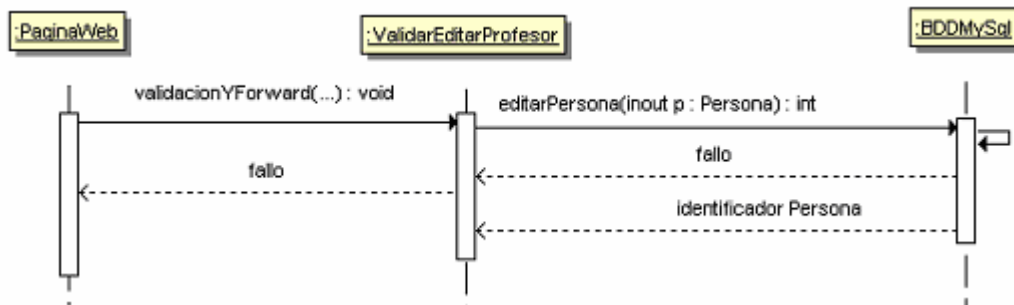
CU – 03 Añadir Personal



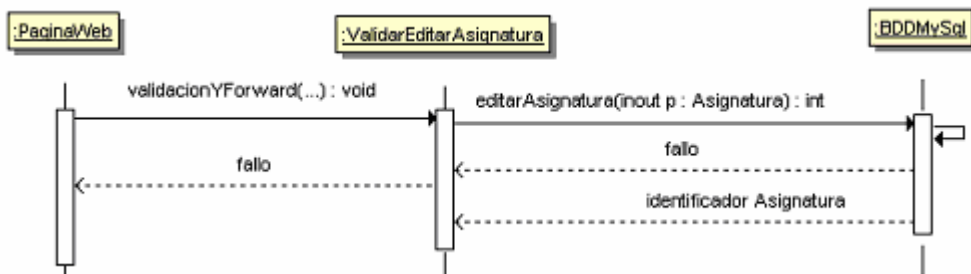
CU – 04 Añadir Asignaturas



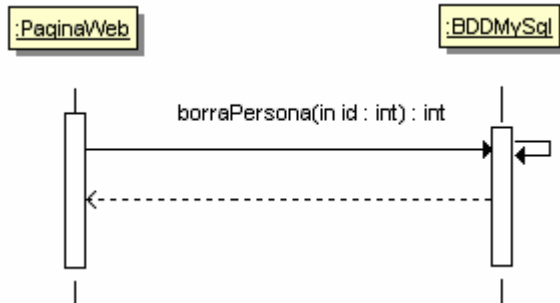
CU – 05 Editar Personal



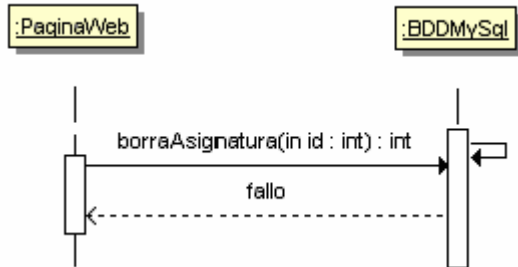
CU – 06 Editar Asignaturas



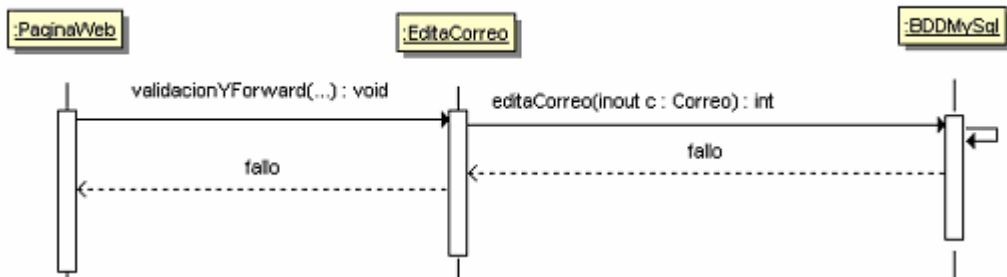
CU – 07 Borrar Personal



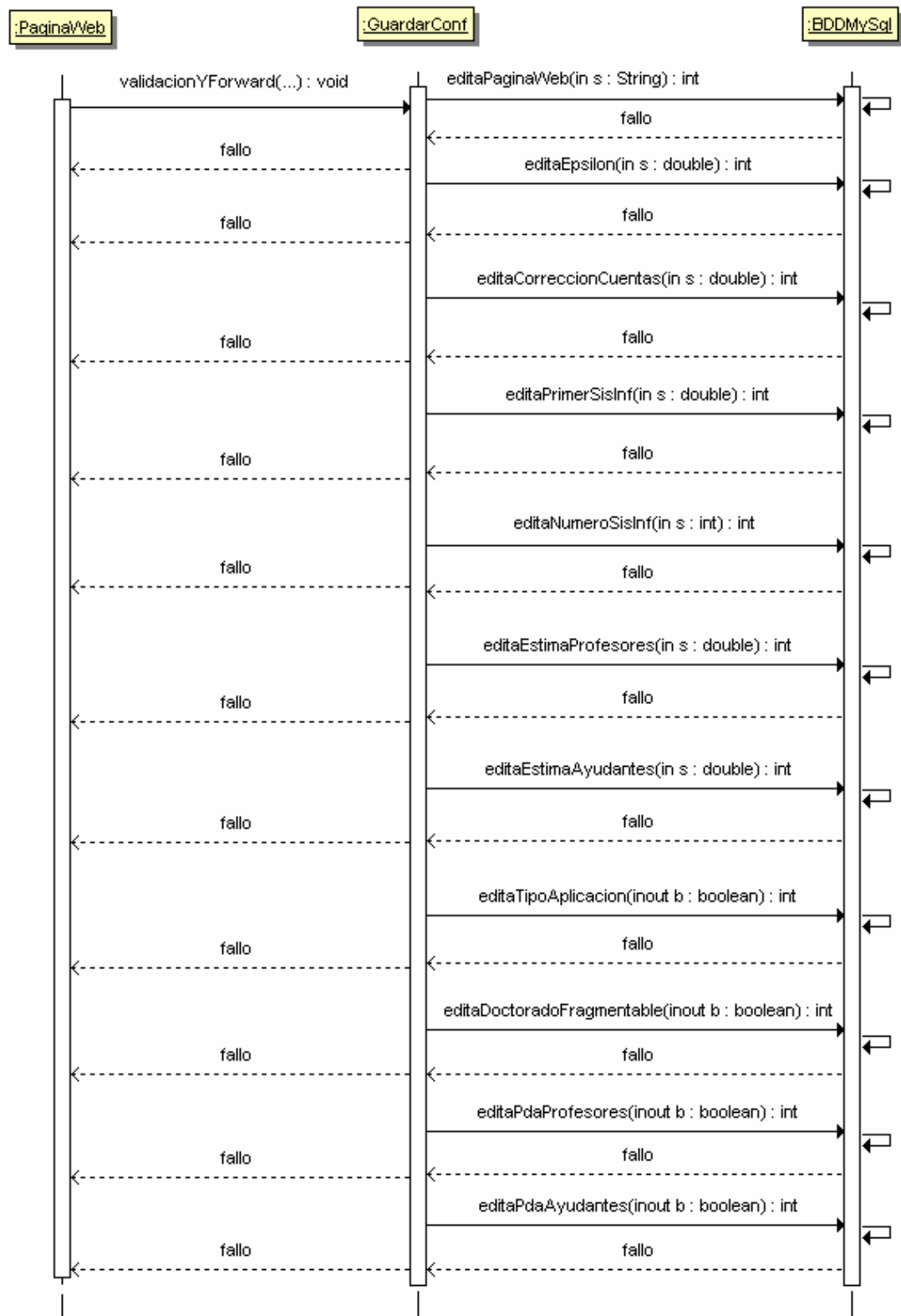
CU – 08 Borrar Asignaturas



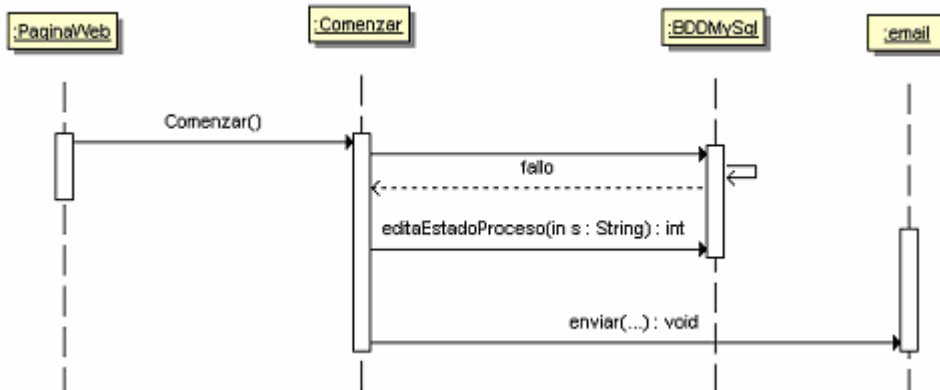
CU – 09 Configurar cuenta de correo



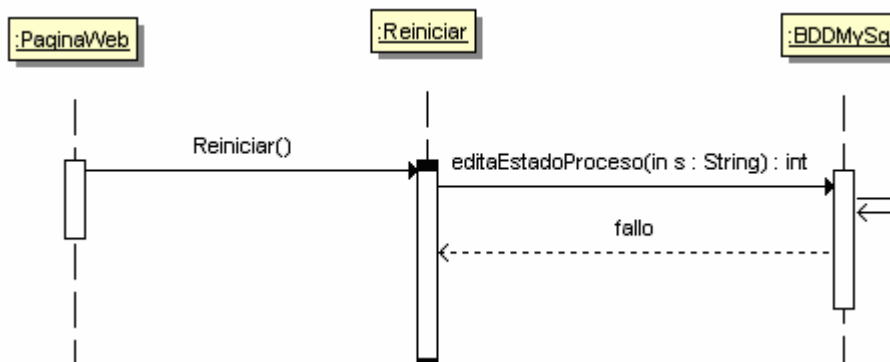
CU – 10 Configurar Aplicación



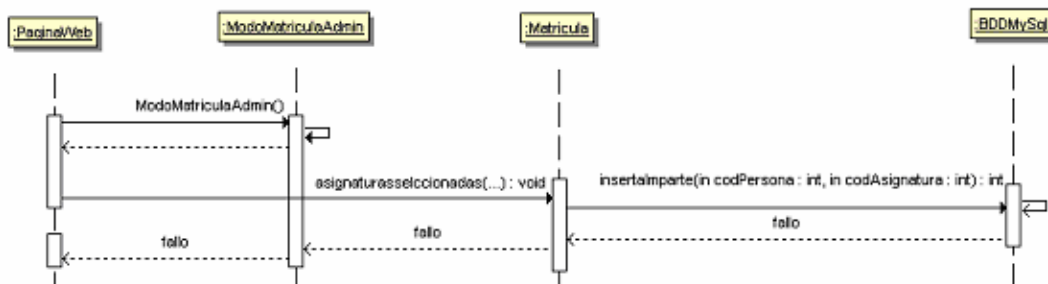
CU – 11 Comenzar proceso de elección de docencia



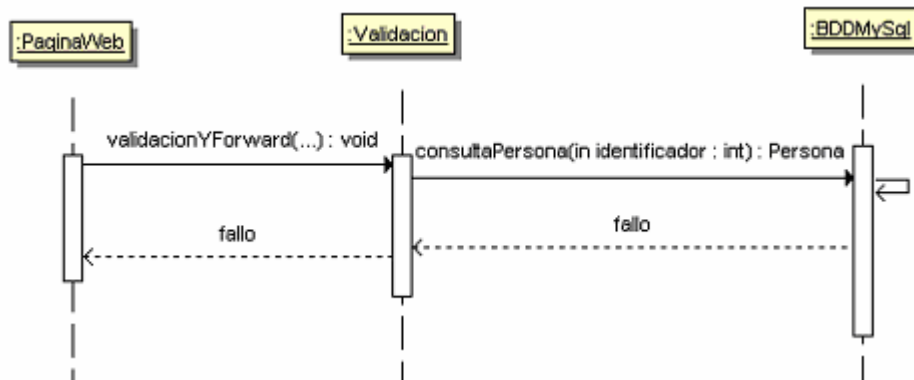
CU – 12 Reiniciar el proceso de elección de docencia



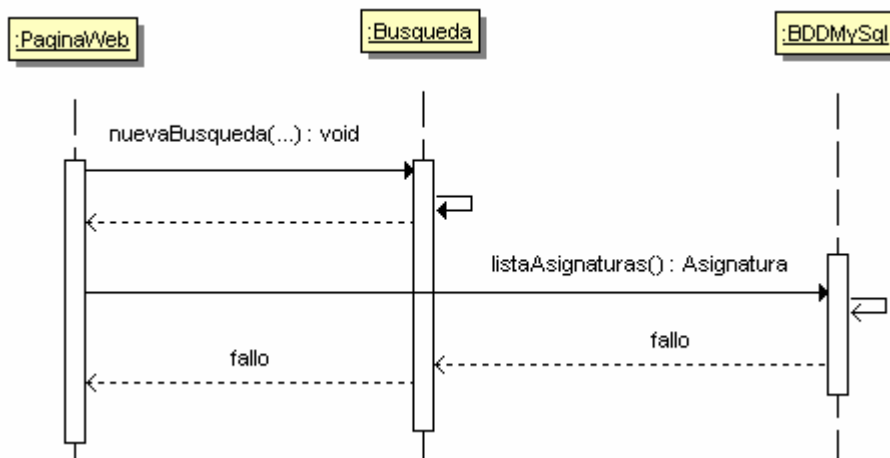
CU – 13 Matricular a alguien en su nombre



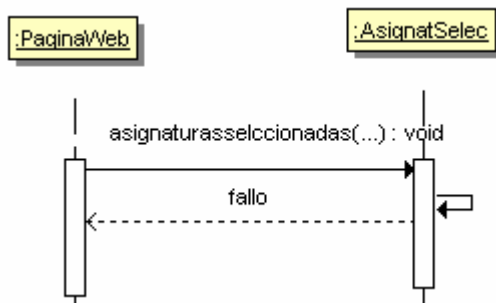
CU – 14 Acceder a la aplicación



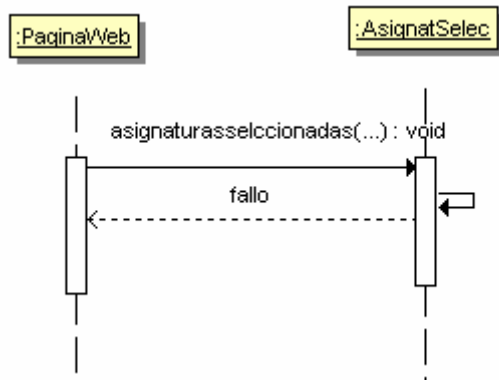
CU – 15 Buscar asignatura selectivamente



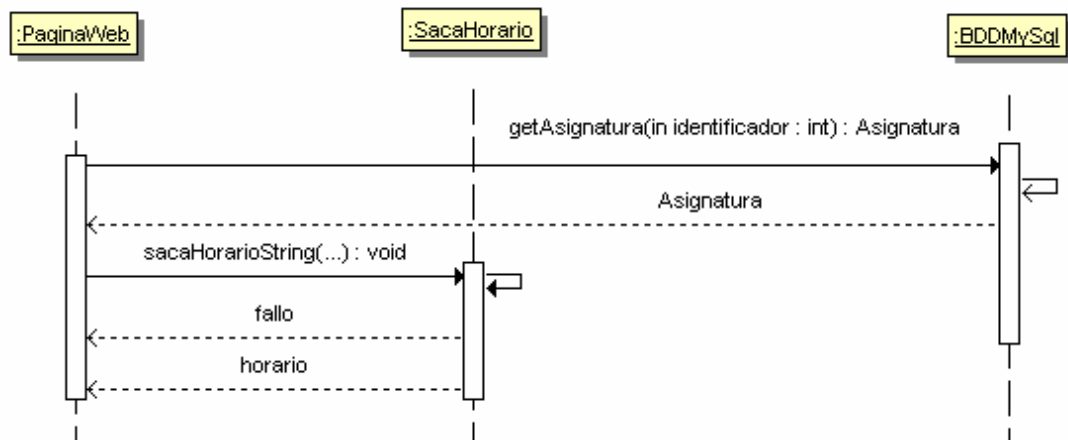
CU – 16 Seleccionar asignatura



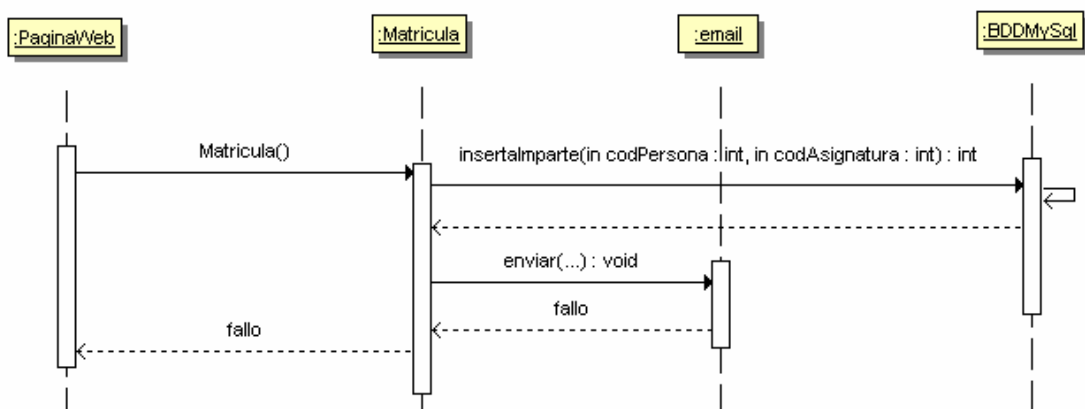
CU – 17 Quitar asignatura



CU – 18 Comprobar horario



CU – 19 Matricularse



3.10. Pruebas de sistema

Para probar la aplicación se ha simulado la asignación de docencia que ha realizado el departamento SIC este mismo año, durante este mes de mayo.

En un principio intentamos realizar las pruebas in situ, durante la reunión que los miembros del departamento llevan a cabo con ese fin. En ellas pudimos apreciar la velocidad con la que llevan a cabo el proceso, sobre todo para los primeros en elegir. Nos fue realmente imposible seguir su ritmo de matrícula, ya que nuestra aplicación no estaba diseñada en aquel momento para esa situación. Sólo estaba implementada una forma de matrícula online, mediante el envío de email al siguiente en elegir, que nos hacía ir muy lentamente. Aun así pudimos tomar contacto con la forma real en la que ellos realizan este proceso, observando posibles puntos de conflictos en nuestro programa pero sobre todo confirmando que nuestra aflicción iba bien encaminada. Tras esta toma de contacto, y después de hablarlo con nuestro director de proyecto, vimos la necesidad de incluir una nueva forma de asignar la docencia, más acorde con la forma actual de realizarse. Esta será desarrollada por un administrador, que podrá ir matriculando uno a uno a los miembros del departamento, sin envío de emails y de manera rápida y sencilla, simulando el proceso de elección de docencia que existe actualmente.

Nuestras segundas pruebas, se realizaron ya una vez el proceso había acabado, usando documentos que nos informasen del desarrollo de la asignación de docencia en cada paso y comprobando que nuestra aplicación se comportaba exactamente como debía en cada uno de estos pasos.

Con estos documentos hubo que ajustar todos los datos, ya que nuestros archivos anteriores estaban desfasados.

- Hubo que eliminar de los archivos relativos a los profesores todos aquellos que por alguna razón este año no van a desarrollar ninguna actividad docente. Hay que asegurarse que únicamente se cargan en el sistema los profesores van a formar parte de la asignación docente.
- Asegurarse que los datos de exención por PDA de profesores y ayudantes son correctos si este dato se va a tener en cuenta en la asignación docente.
- Asegurarse que los horarios están en la última versión, ya que sino podrían darse casos de falso solapamiento de horarios.
- Nos dimos cuenta de una labor que debe llevar a cabo el encargado de introducir los datos en el sistema. Existen algunas asignaturas que son asimétricas (poseen un cuatrimestre con más créditos que el otro) y que deberán ser actualizadas manualmente, ya que por defecto nuestra aplicación divide las asignaturas anuales de forma equitativa.

Con estos cambios, y teniendo en cuenta la previsión de profesores a incorporarse el año que viene, las cuentas cuadran de principio a fin, y se obtiene como salida un archivo Excel con una estructura exacta a la real obtenida este año en el departamento.

4. Conclusiones

Nuestro proyecto ha sido el resultado de un esfuerzo en diversos campos, para obtener una herramienta que no sólo realice el proceso de matriculación del departamento, si no que también lo facilite, tanto para el usuario final, como para el encargado de administrar el proceso.

El primero de los esfuerzos mencionados fue el de la investigación, tanto de las tecnologías a emplear, como en el del tipo de solución concreta que íbamos a ofrecer. Este período nos llevó más tiempo del esperado, principalmente por que todos los conocimientos que teníamos sobre el desarrollo de aplicaciones Web eran muy teóricos y superficiales, siendo en muchos casos nulos.

Como resultado se decidió que lo mejor sería hacer un desarrollo basado en JSP, por ser nuestro conocimiento en las tecnologías Java el más amplio, permitiéndonos así tener un control pleno sobre todo tipo de cálculos y validaciones, así como en la información presentada al usuario. Esto último no es posible en otras tecnologías que se barajaron, como pueda ser .NET o PHP, pero sin embargo estas destacaban en la simplicidad del desarrollo. Decisión difícil no tener todo en uno.

Nuestra aplicación destaca principalmente en la comodidad que puede obtener el usuario final de la matriculación a la hora de realizarla, pudiendo elegir a distancia, de una forma cómoda, y sin horario estricto, lo cual era un objetivo clave. También ofrece al encargado de administrar el proceso una serie de ventajas, como pueda ser la modificación de cualquiera de los datos, o una visualización de estos mismos de una forma sencilla y ordenada. Sin embargo el principal inconveniente llega a la hora de establecer validaciones sobre el proceso, ya que nuestro sistema es más estricto en este aspecto. Es decir, no se tiene la facilidad de ‘ajustar’ datos que tiene la forma clásica en la que el proceso se ha estado realizando siempre.

Como resultado de las pruebas finales, y conversaciones con el profesor director de este proyecto es el resultado de otro de los modos de matriculación, el realizado íntegramente por el administrador, u otra persona encargada de ello. Básicamente este proceso sustituye las anotaciones manuales que en las reuniones se realizaban, y realiza todos los cálculos y validaciones sobre la marcha, pero en ningún caso sustituye la reunión entre los profesores del departamento.

La última manera de realizar el proceso de matrícula consiste en la automatización de ésta mediante la petición a los profesores de una serie de asignaturas favoritas. El proceso pretende siempre intentar satisfacer las peticiones del mayor número de profesores, pero dado que en la realidad, este es grande, y el número de asignaturas también es alto, el proceso no encuentra una matriculación en la que todos los profesores tengan asignaturas de entre sus favoritas. Este problema no se daría en el caso de que los profesores elijan unas asignaturas variadas, o tengan muchas favoritas, lo cual en un proceso real nunca se encuentra.

5. Manual del usuario


1. Identificarse en la aplicación
2. Vista general de la Web
3. Menú de opciones
4. Funcionalidades de la aplicación
 - a. Administración
 - i. Registro de asignaturas/profesores
 - ii. Configuración de la aplicación
 - b. Matriculación usuarios
 - i. Proceso individual
 - ii. Matriculación por el administrador
 - iii. Matriculación automática
5. Instalación de la aplicación

Identificarse en la aplicación



Nada más conectarse, el usuario deberá introducir su número de identificación y la contraseña. Estos datos habrán sido enviados por correo por la propia aplicación en la mayoría de los casos, siempre que el correo del usuario esté registrado.

Visualizar: Sólo texto | Normal | Grande | Pequeño | Alto contraste

Lunes, 9 de junio de 2008



Departamento de Sistemas
Informáticos y Computación
Universidad Complutense de Madrid



Usuario
Conectado como:
Visitante

Elección de docencia Sic

Debe identificarse primero para acceder a todas las funciones

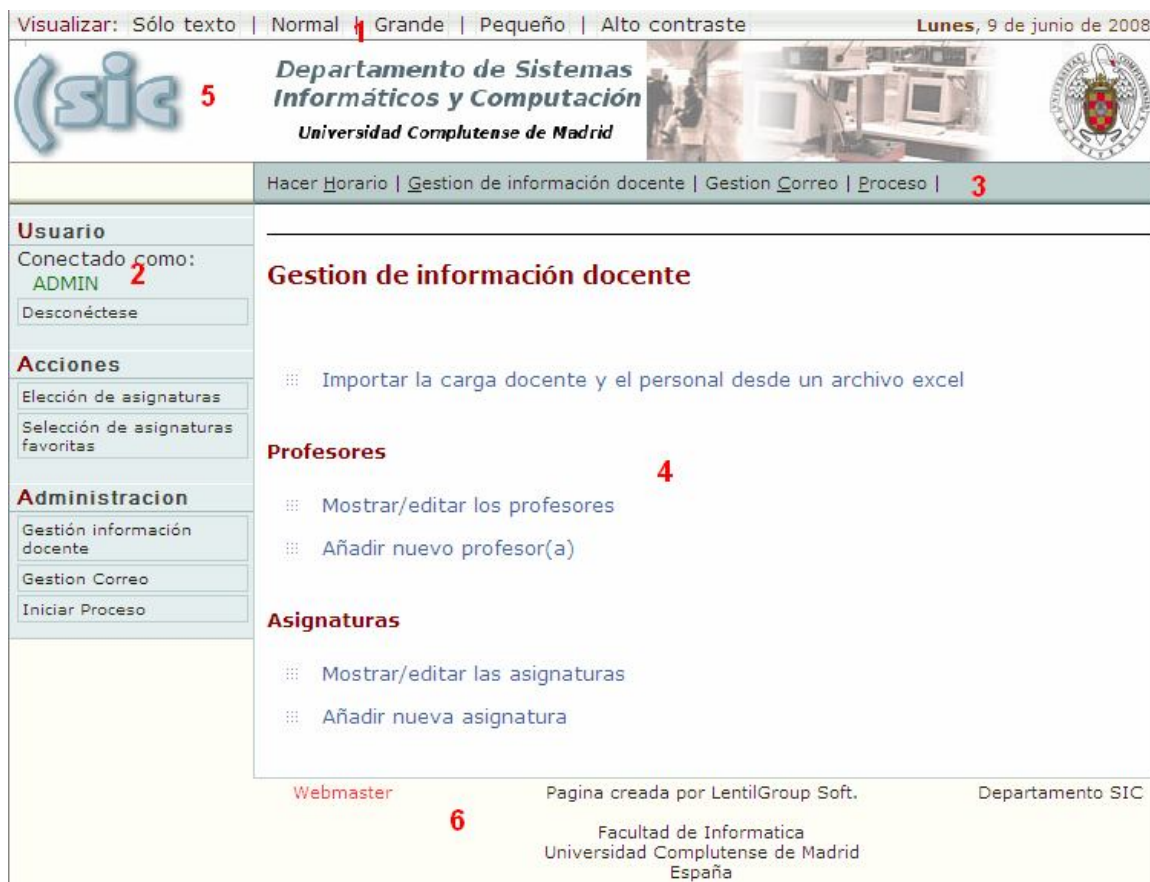
Por favor, identifique con su número y contraseña de usuario

*Número de usuario:

*Contraseña:

Vista general de la Web

La página Web estará en todo momento dividida en secciones fácilmente identificables. Cada una con un contenido propio. Son las siguientes.



Visualizar: Sólo texto | Normal | **1** Grande | Pequeño | Alto contraste Lunes, 9 de junio de 2008

(sic) 5 Departamento de Sistemas Informáticos y Computación
Universidad Complutense de Madrid

Hacer Horario | Gestion de información docente | Gestion Correo | Proceso | **3**

Usuario
Conectado como: **ADMIN 2**
Desconéctese

Acciones
Elección de asignaturas
Selección de asignaturas favoritas

Administración
Gestión información docente
Gestion Correo
Iniciar Proceso

Gestion de información docente

Importar la carga docente y el personal desde un archivo excel

Profesores 4
Mostrar/editar los profesores
Añadir nuevo profesor(a)

Asignaturas
Mostrar/editar las asignaturas
Añadir nueva asignatura

Webmaster **6** Pagina creada por LentilGroup Soft. Departamento SIC
Facultad de Informatica
Universidad Complutense de Madrid
España

i. Menú de estilo/accesibilidad

Permite en todo momento seleccionar el estilo en el que se verá la Web, lo que cambiará, entre otras cosas, el tamaño de la fuente, o el contraste general de la página, permitiendo a ésta ser utilizada por usuarios con deficiencias visuales.

ii. Menú propio del usuario

En este menú se mostraran en todo momento los datos del usuario conectado, ofreciéndole la posibilidad de una desconexión segura. Dispondrá además de las opciones para las que tenga permiso, que variarán si es un administrador de la aplicación o no.

iii. Menú rápido

Ofrece al usuario los enlaces más comunes para poder navegar rápidamente por la Web.

iv. Bloque central

Muestra el contenido seleccionado por el usuario, o bloques lanzados por la página, como pueda ser el diálogo de registro, un listado de asignaturas, etc.

v. Logos

Aquí se muestra en todo momento el logo del departamento así como el de la Universidad, siendo estas imágenes enlaces a sus respectivas páginas.

vi. Pie de página

Mostrará información acerca de la página, y además la posibilidad de contactar con el departamento encargado de la aplicación vía e-mail.

Menú de opciones

Para un usuario normal:

Información del usuario conectado, y desconexión segura. Así como las acciones que se le permiten hacer, siendo estas la elección de asignaturas, o la selección de sus asignaturas favoritas, para un posible proceso automático.

Usuario
Conectado como: Mario Rodríguez Artalejo
<input type="button" value="Desconéctese"/>
Acciones
<input type="button" value="Elección de
asignaturas"/>

Para un administrador:

Además de lo mostrado para un usuario, están las utilidades de gestión, más adelante comentadas.

Usuario
Conectado como: ADMIN
<input type="button" value="Desconéctese"/>
Acciones
<input type="button" value="Elección de
asignaturas"/>
Administracion
<input type="button" value="Gestión información
docente"/>
<input type="button" value="Gestion Correo"/>
<input type="button" value="Iniciar Proceso"/>

Funcionalidades de la aplicación

Administración

Registro de asignaturas/profesores

Registro de nuevas asignaturas

Para añadir una asignatura se comprobará que todos los campos necesarios estén rellenos, así como la validez de los datos. Es decir, la aplicación informará al usuario si, por ejemplo, intenta meter un número de créditos de ‘trece’ en vez de ‘13’. La mayoría de campos aceptan cualquier cadena de caracteres como entrada, a excepción del número de créditos que deben ser números. El horario también necesita de un formato especial, siendo este el mismo utilizado por la facultad en su Web, i.e. una letra para el día, o varias separadas por guiones para indicar más de un día seguida de dos puntos, y el rango de horas separados por un guión. Además se puede añadir otra regla de días separadas la conjunción ‘y’. Un ejemplo clarificador es el que sigue: “*L,M,V:18-19 y J:17-18*”. El campo Profesor/Ayudante debe poseer un String que sea o ‘profesor’ o ‘ayudante’, para marcar a quien va dirigida esta asignatura. Rellenar este campo es importante ya que ambos procesos son totalmente independientes, y podría influir en que las cuentas no cuadrasen, por ejemplo.

Datos para la nueva asignatura:

Nombre:	Centro:	Titulación:	Curso:	Cuatr.:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Grupo:	Prof/Ayudant:	Creditos:	Horario:	Edificio:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Aula:	Profesor Asignado:	Fragmentable:		
<input type="text"/>	<input type="text" value="Ninguno"/>	<input type="text" value="No"/>	<input type="button" value="Insertar"/>	

Edición de asignaturas

Se realiza con las mismas validaciones que antes, pero para una asignatura ya creada. Desde aquí también se pueden eliminar asignaturas, lo cual no es una práctica recomendada.

Listado de las asignaturas:

	Nombre	Centro	Titulación	Curso	Cuatr.	Grupo	Prof/Ayudante	Creditos	Horario	Edificio	Aula
<div>Editar</div> <div>Eliminar</div>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	A	profesor	4.5			
<div>Editar</div> <div>Eliminar</div>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	A	profesor	4.5			
<div>Editar</div> <div>Eliminar</div>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	B	profesor	4.5			
<div>Editar</div> <div>Eliminar</div>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	B	profesor	4.5			

Registro de nuevos profesores

Para añadir una persona nueva se comprobará que todos los campos necesarios estén rellenos, así como la validez de los datos.

Los campos aceptan cadenas de caracteres, a excepción de la categoría y los créditos que deben ser números, en el caso de la categoría un número entero. Al igual que en las asignaturas, el campo Profesor/Ayudante debe ser relleno con el puesto de la persona.

Datos para la nueva persona a añadir:

Nombre:	Contraseña:	Categoría:	Mail:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Profesor Ayudante:	Es administrador:	Créditos PDA:	
<input type="text"/>	<input type="text" value="No"/>	<input type="text"/>	
<div>Insertar</div>			

Edición de profesores

Se realiza con las mismas validaciones que antes, pero para una persona ya introducida en el sistema. Desde aquí también se pueden eliminar a las personas de la base de datos.

Listado del personal:

	Nombre	Contraseña	Categoría	e-Mail	Es administrador:	Profesor Ayudante:	Créditos PDA:	Matriculado:
Editar Eliminar	Mario Rodríguez Artalejo	DDh7F6Ef	1	pruebassic@gmail.com	No	profesor	0.0	No
Editar Eliminar	Ricardo Peña Marí	9AB6a5jj	2	pruebassic@gmail.com	No	profesor	0.0	No
Editar Eliminar	Narciso Martí Oliet	C498FI6C	3	pruebassic@gmail.com	No	profesor	0.0	No
Editar Eliminar	Teresa Hortalá González	i1f8Gh4C	4	pruebassic@gmail.com	No	profesor	0.0	No
Editar								

Configuración de la aplicación

Importación de los ficheros Excel con la información relativa al proceso.

La manera más cómoda de introducir tanto a los profesores como las asignaturas en el sistema es importándolos desde un fichero de MS Excel. Para el funcionamiento de la aplicación se deben importar los archivos Excel que contengan la información relativa a los profesores y a las asignaturas; además de éstos habrá que importar el fichero que contenga la información sobre los créditos de PDA de profesores y ayudantes. El formato que ha de tener cada uno de ellos es el siguiente:

Excel Asignaturas

El archivo deberá contener una pestaña cuyo nombre ha de contener la palabra “carga” (no distingue entre mayúsculas y minúsculas por lo que podremos escribirlo en el formato que deseamos), de esta manera el programa detectará que ahí es donde se encuentra la información que queremos introducir en la base de datos.

Deberá contener una serie de columnas con la información situadas en el siguiente orden: Centro, Curso, Título, Asignatura, Grupo, Cuatrimestre, Créditos teóricos de la Asignatura, Créditos Prácticos, Desglose, Total de créditos (que corresponde a los créditos prácticos por el desglose más los teóricos). Si a continuación se añade alguna columna más, no afectará al funcionamiento del programa.

Otra cosa a tener en cuenta son los colores que le apliquemos a las distintas celdas. Es decir, si a una fila le aplicamos el color azul turquesa significa que es un laboratorio por lo que la aplicación lo tratará como tal, es decir, separando los créditos en teóricos y prácticos del profesor y creando los ayudantes que hayamos tenido en cuenta a la hora de escribir el desglose (desglose = profesor + número ayudantes).

Aparecerán en gris todas aquellas filas que contengan créditos por situaciones especiales como pueden ser exenciones y suplencias.

En este último caso, hay que destacar otro aspecto del formato y es que si se desea que el programa lea el nombre del profesor al que va asignado y lo trate se deberá poner entre corchetes; es decir, si un cierto profesor (Juan Sánchez Pérez) ha realizado una suplencia por otro lo pondremos así: Suplencia por Marta Alonso [Juan Sánchez Pérez]. Podemos poner todo el texto que queramos, pero si hay que asignarle a ese profesor los créditos ya realizados el nombre deberá aparecer de la forma que acabamos de indicar.

Teniendo en cuenta todos estos detalles, obtendremos una hoja en Excel muy similar a la que se muestra a continuación como ejemplo:

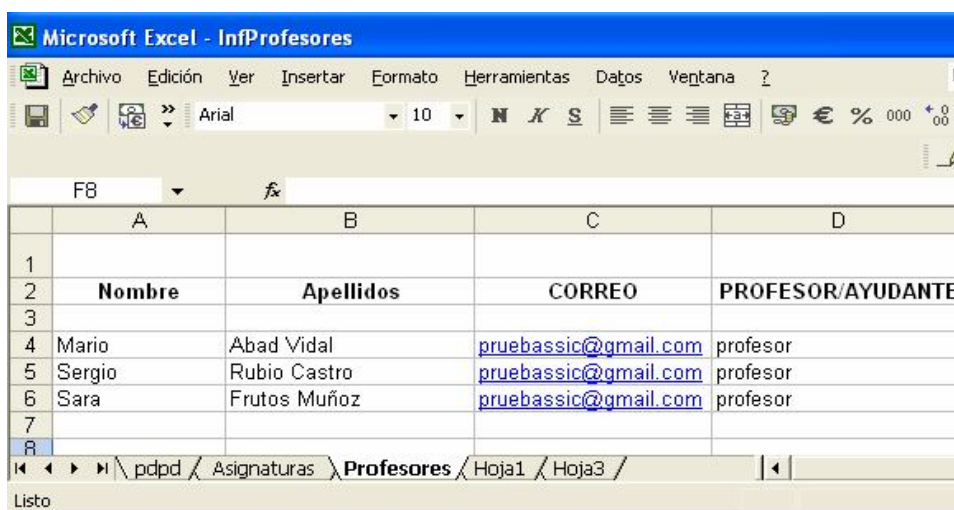
Microsoft Excel - Plan docente 2008													
Archivo Edición Ver Insertar Formato Herramientas Datos Ventana ?													
Escriba una pregunta													
A1 * Carga docente para el curso 08-09													
A	B	C	D	E	F	G	H	I	J	K	L	M	N
5													
6													
7													
8													
9	FI	1º	Ing. Téc. en Inf. de Gestión	Lógica	Grp	Al.	Cuat.	C.T.	C.P.	Des.	Total	Ayud	Titulación
10	FI	1º	Ing. Téc. en Inf. de Gestión	Lógica	A	86	2	3	1,5	1	4,5	0	
11	FI	Opt.	Ing. Téc. en Inf. de Gestión	Programación funcional	B	43	2	3	1,5	1	4,5	0	
12	FI	1º	Ing. Téc. en Inf. de Sistemas	Lógica	A	28	1	3	1,5	1	4,5	0	13,5
13	FI	2º	Ing. Téc. en Inf. de Sistemas	Estructura de datos y de la información	A	100	2	3	1,5	1	4,5	0	
14	FI	2º	Ing. Téc. en Inf. de Sistemas	Laboratorio de programación II	B	68		9	3	1	12	0	
15	FI	2º	Ing. Téc. en Inf. de Sistemas	Laboratorio de programación II	A	85		3	6	2	15	6	
16	FI	Opt.	Ing. Téc. en Inf. de Sistemas	Programación funcional	B	51		3	6	2	15	6	
17	FI	1º	Ingeniería en Informática	Introducción a la programación (total: 323)	A	32	1	3	1,5	1	4,5	0	51,0
18	FI	1º	Ingeniería en Informática	Introducción a la programación	A	80		6	3	1	9	0	
19	FI	1º	Ingeniería en Informática	Introducción a la programación	B	80		6	3	1	9	0	
20	FI	1º	Ingeniería en Informática	Laboratorio de programación I (total: 277)	C	80		6	3	1	9	0	
21	FI	1º	Ingeniería en Informática	Laboratorio de programación I	A	80	2	1,5	3	2	7,5	3	
22	FI	1º	Ingeniería en Informática	Laboratorio de programación I	B	58	2	1,5	3	2	7,5	3	
23	FI	Gen.	Ingeniería en Informática	Fundamentos informáticos del arte contemporáneo	C	80	2	1,5	3	2	7,5	3	
24	FI	Sem.	Ingeniería en Informática	Introducción a los sistemas multi-agentes	A	51	1	4,5	0	1	4,5	0	
25	FI	Sem.	Ingeniería en Informática	Accesibilidad en la web	A	20	2	4,5	0	1	4,5	0	
26	FI	Sem.	Ingeniería en Informática	Algoritmos para realizar búsquedas en la web	A	81	1	0	4,5	2	9	4,5	
27				Exenciones por cargo académico	A	53	1	1,5	3	2	7,5	3	75,0
28				Vicedecano (12cr) [Teresa ortalá González]							10		10
29				Vicedecano (12cr) [Fernando Rubio Díez]							10		10
30				Secretario de departamento (6cr) (3cr para sumar PDA completo)[J. Alberto Verdejo López]							3		3
31				Director de Departamento (6cr) (3cr para sumar PDA completo) [Mario Rodríguez Artalejo]							3		3
32				Subdirector de departamento (6cr)[Ismael Rodríguez Laguna]							6		6
33				Exención sindical 12cr (Francisco Crespo)[Francisco J. Crespo Yáñez]							10		10
34				Suplencias									
35				(supl NLB) Crespo, Francisco J. [Francisco J. Crespo Yáñez]							4,5		4,5
36				(supl NLB) Rodríguez, Ismael[Ismael Rodríguez Laguna]							4,5		4,5
37		3º ciclo	Doctorado	Extensiones de programación lógica	1			4	0	1	4		
38		3º ciclo	Doctorado	Deducción automática	1			2	0	1	2		
39		3º ciclo	Doctorado	Especificación y validación de software	1			5	0	1	5		
Carga Docente NC + GSME 08-09 / Suplencias para el curso 08-09 / PD Asignaturas /					4								

Excel Profesores

El archivo que contiene los datos de los profesores debe tener una pestaña que ponga “profesor” o derivados, es decir, títulos que contengan al menos ese nombre (por ejemplo: lista profesores, profesorado...), sin embargo da igual si está escrito en mayúsculas o minúsculas.

Al igual que ocurre en la hoja de las asignaturas, esta hoja donde aparecen los profesores ha de tener en una primera columna el nombre del profesor (cuyo nombre ha de ser “nombre” o un derivado), en la siguiente los apellidos, a continuación el mail, y, por último, indicar si es un profesor o un ayudante.

El formato que obtendremos será similar a este:



	A	B	C	D
	Nombre	Apellidos	CORREO	PROFESOR/AYUDANTE
4	Mario	Abad Vidal	pruebassic@gmail.com	profesor
5	Sergio	Rubio Castro	pruebassic@gmail.com	profesor
6	Sara	Frutos Muñoz	pruebassic@gmail.com	profesor

Excel PDA

Para que el programa reconozca la hoja, de la que tenemos que leer los créditos por méritos especiales que obtienen los profesores y ayudantes, ésta deberá tener el nombre “pda”.

Los créditos que el programa tendrá en cuenta son los introducidos en la parte de núcleo central. Para identificar este apartado deberemos poner un título que contenga la frase “sólo núcleo” y leerá hasta el siguiente título que es el comienzo del siguiente bloque.

Las columnas que no pueden faltar son (en este orden): Categoría, Nombre, Apellidos, Créditos, PDA. Es imprescindible que todos y cada uno de estos campos estén rellenos.

El archivo que contenga los créditos PDA ha de tener una hoja con un formato similar al mostrado en la siguiente figura:

Microsoft Excel - PDA 08-09

Archivo Edición Ver Insertar Formato Herramientas Datos Ventana ?

Book Antigua 9

D86 fx 1

	A	B	C	D	E	F	G	H	I	J
90	SOLO NÚCLEO CENTRAL:									
91										
92	Profesores (TC y TP):									
93	Categoría	Nombre	Apellidos	CR	PDA	Var	Trunc	Var	CR/2	PDA/2
94	CU	Silvia	Gil Arano	12,8	9	-8,26	4,5	-4,03	6,4	3
95	CU	Rafael	Martín Toledo	8,6	8,6	-7,86	4,5	-4,03	4,3	4,3
96	CU	Noelia	Galán Suarez	5,3	5,3	-4,56	4,5	-4,03	2,65	2,65
97	TU	Cristina	Sousa Cabrera	9,3	3	-3	3	-3	4,65	3
98	Total			36	25,9		16,5		18	12,95
99	Media			1,03	0,74		0,47143		0,5143	0,37
100										
101	Ayudantes:									
102	Categoría	Nombre	Apellidos	CR	PDA	Var	Trunc	Var	CR/2	PDA/2
103	Ayudante	Antonio	Hernández Ramos	0,5	0,5	-0,16	0,5	-0,3	0,25	0,25
104	Ayudante	Roberto	Torres de Alba	0,2	0,2	0,14	0,2	0	0,1	0,1
105	Ayudante	Susana	Rojo Medina	3	3	-2,66	1,5	-1,3	1,5	1,5
106	Total AY			3,7	3,7		2,2		1,85	1,85
107	Media AY			0,34	0,336		0,2		0,1682	0,16818
108										
109	SÓLO GSME:									
110										

Lista

Si durante la importación se encuentran errores en el fichero, como puedan ser los nombres de los profesores en las asignaturas, la aplicación tratará de corregirlos, informando de ello al usuario:

Se completó la subida del fichero correctamente.

Ahora puede introducir los datos, o corregir el fichero y volverlo a subir.

Introducir los datos en el sistema

Volver atrás

Se encontraron los siguientes errores en el archivo.

El nombre del profesor de la asignatura **Vicedecano [Teresa ortalá González]** no se encontró en la base de datos. Quizá quiso decir: **Teresa Hortalá González**

Cambiar por el sugerido

Cambiar por el seleccionado

Teresa Hortalá González

Es necesario subir los ficheros en el orden marcado, así pues la aplicación no dejará subir o bajar un archivo concreto mientras no tenga las dependencias previas satisfechas.

Selecciona el fichero con la información de los profesores (formato excel) :

Subido

Selecciona el fichero con la información de las Asignaturas (formato excel) :

Subido

Selecciona el fichero con los créditos de PDA (formato excel) :

Una vez importado todo descargue el fichero con las asignaturas :

Selecciona el fichero devuelto por el sistema, con los datos ya rellenos :

Configuración de los datos de la cuenta de correo.

Necesitamos marcar ciertos datos de la cuenta de correo desde la que se realizarán los envíos de diversa información a los profesores, tales como el nombre de la cuenta, la contraseña, el servidor y el puerto, así como las medidas de seguridad de esta, y un nombre para mostrar. Desde esta misma pantalla se puede probar si los datos introducidos son correctos.

Los parámetros a configurar son:

- Nombre a mostrar: Nombre que tendrá el remitente de los mensajes enviados
- Cuenta y Contraseña: Del que realizará el envío de los emails.
- Servidor SMTP: a través del cual se realiza el envío de mail. La cuenta usada debe tener este servicio disponible.
- Puerto: habitualmente es el 25, pero puede variar.
- TLS y Autenticación: Protocolos de seguridad a distintos niveles. Recomendamos activarlos siempre.

Para mayor comodidad incluimos las configuraciones predeterminadas de algunos servicios de correo habituales.

Puede probar si la configuración introducida es correcta con el botón PROBAR, que enviará un correo electrónico a la propia cuenta. Si no se recibe el correo evidentemente algo está funcionando mal.

Nota: Muchos servicios de correo requieren algún tipo de activación para usar el correo a través de Pop3 y Smtplib, ya que suelen operar únicamente a través de webmail. Es el caso por ejemplo de Yahoo.

Gestion de Correo

Desde aquí se gestiona la cuenta de Correo que mandará toda la información a los participantes del proceso de elección de docencia del departamento.

Configuraciones Predeterminadas : Ninguna ▼

Nombre a Mostrar : ELECCION DE DOCENCIA SIC	
Cuenta : pruebassic@gmail.com	Contraseña : ●●●●●●
Servidor SMTP : smtp.gmail.com	Puerto : 587
TLS : Si ▼	Autenticacion : Si ▼

Guardar

Prueba el correcto funcionamiento de la configuración de correo.
Esto enviará un mail a tu misma cuenta.
GUARDA ANTES LA CONFIGURACION DE CORREO.

Probar

Parámetros del proceso de matrícula.

El proceso de matrícula necesita de una serie de parámetros o constantes, que pueden variar de un año a otro, por eso se ofrece la posibilidad de configurar sus valores.

- Dirección pagina Web: contiene la dirección Web donde se encuentra alojada la aplicación, para poder comunicársela a los participantes y que puedan acceder a ella.
- epsilon: valor necesario para el cálculo de valor de créditos mínimo para matricularse.
- Incentivo primera asignatura SI: incentivo a la primera asignatura de sistemas informáticos que se ha matriculado.
- Número de grupos de SI: cantidad máxima de asignaturas de sistemas informáticos que se le permite matricular a un profesor.

- Fragmentación máxima: máximo número de créditos que se le permite coger a un profesor cuando fragmenta una asignatura.
- Doctorados fragmentables: para considerar a todas las asignaturas de doctorado como fragmentables o no.
- Créditos PDA: para marcar si se tendrá en cuenta la exención por PDA en la asignación docente de los profesores.
- Créditos PDA ayudantes: para marcar si se tendrá en cuenta la exención por PDA en la asignación docente de los ayudantes.
- Estimación profesores: estimación para los profesores contratados para el año que viene. Influye en las cuentas.
- Estimación ayudantes: estimación para los ayudantes contratados para el año que viene. Influye en las cuentas.
- CorreccionCuentas: Al cálculo de los créditos totales para contabilizar se suelen hacer correcciones debido a asignaturas que no las dará el departamento, y que se pasaran por ejemplo al departamento GSME. Aquí deberá ir el total de la suma de todos ellos y el programa se encargará de restárselo al total.

Iniciar Proceso

Se va a proceder a empezar el proceso de matriculación del departamento Sic de la UCM

Asegúrese de que ha realizado los siguientes pasos :

1. Ha cargado el archivo excel con la información docente necesaria. Sino ir [aquí](#).
2. Ha verificado que la información cargada es correcta.
3. Ha rellenado los datos del Correo y ha verificado su funcionamiento. Sino ir [aquí](#).

Relleno estos campos de configuración y pulse Comenzar. Se enviará entonces a todos los participantes un mail con la información necesaria y dará comienzo la elección de docencia.

Rellene este campo con la dirección donde se encuentra alojada esta aplicación web, para así enviársela a todos los usuarios de la aplicación.	
Dirección Web : <input type="text"/>	
El mínimo a matricular se calcula por la operación (Nº créditos pendientes) / (Nº personas pendientes) - Epsilon .	
Epsilon : <input type="text"/>	
Hay 38 profesores en la aplicación, añada la estimación de los profesores a incorporarse el año que viene : <input type="text"/>	
Hay 12 ayudantes en la aplicación, añada la estimación de los ayudantes a incorporarse el año que viene : <input type="text"/>	
Consideraciones con la asignatura de Sistemas Informáticos	
Incentivo por la primera asignatura de SI : <input type="text"/>	Maximo número de asignaturas de SI que se pueden impartir: <input type="text"/>
Asignaturas de Doctorado	
Considerar asignaturas de Doctorado como fragmentables. <input checked="" type="checkbox"/>	Maximo número de créditos permitidos al fragmentar un Doctorado: <input type="text"/>

Matriculación usuarios

Proceso individual

El profesor con el turno para matricularse puede elegir las asignaturas de la tabla inferior de la pantalla. En la parte superior aparece toda clase de información que concierna a su elección docente:

- El mínimo número de créditos que debe matricular
- Cuantos créditos lleva ya matriculados
- Las asignaturas elegidas con su información básica y la posibilidad de deseccionarlas.
- El horario de las asignaturas elegidas dividido por cuatrimestres.

Si alguna elemento del reglamento de matrícula se incumple el programa informará de ello justo debajo del botón de matrícula, e impedirá la misma hasta que no se solucione.

Para encontrar asignaturas más fácilmente el usuario dispone de de un conjunto de búsquedas que le permiten acotar la información mostrada en la tabla de selección. Únicamente deberá seleccionar en los menús desplegables el parámetro que desea que se cumpla. Para búsquedas por horario se deberá escribir en el formato anteriormente comentado una franja de días y horas dentro de la cual quiere que estén las asignaturas.

Cuando todo esté listo, con solo pulsar Terminar habrá conseguido matricularse. En breve recibirá un mail con su elección de docencia.

1º Cuatrimestre
2º Cuatrimestre

Créditos mínimos: 19,94 Créditos

Créditos: 12 Créditos

	Nom	Cent	Titul	Curso	Cuatr	Grupo	Credit
<input checked="" type="checkbox"/>	Exención PDA						9.0
<input checked="" type="checkbox"/>	Director de Departamento [Mario Rodríguez Artalejo]						3.0

Terminar

1º cuatrimestre

	Lunes	Martes	Miercoles	Jueves	Viernes
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					
17-18					
18-19					
19-20					
20-21					

Marque las asignaturas que desea seleccionar:

Libres	Centro	Titulación	Curso	Cuatr.	Grupo	Creditos	Horario
Busquedas	To	Tod	Todos	To	To	Todo	

	Nombre	Centro	Titulación	Curso	Cuatr.	Grupo	Prof/Ayudante	Creditos	Horario	Edificio	Aula
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	A	PROFESOR	4.5			
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	A	PROFESOR	4.5			
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	B	PROFESOR	4.5			
	Introducción a		Ingeniería								

Matriculación por el administrador

El proceso de matriculación individual puede ser realizado por el administrador, lo que sería equivalente al proceso que se ha realizado habitualmente en las reuniones, con la salvedad de que la aplicación es la que realiza ahora todos los cambios.

Un administrador sólo debe pulsar en 'Matricular desde cuenta admin' y comenzará a matricular al personal por orden de prioridad.

Usuario
Conectado como:
ADMIN
Desconéctese

Acciones
Elección de asignaturas

Administración
Gestión información docente
Gestión Correo
Iniciar Proceso

Dejar de Matricular

Esta matriculando a :

1º Cuatrimestre
2º Cuatrimestre

Mario Rodríguez Artalejo

1º cuatrimestre

Créditos mínimos: 19,94 Créditos

Créditos: 12 Créditos

	Nom	Cent	Títul	Curso	Cuatr	Grupo	Credit
<input checked="" type="checkbox"/>	Exención PDA						9.0
<input checked="" type="checkbox"/>	Director de Departamento [Mario Rodríguez Artalejo]						3.0

Terminar

	Lunes	Martes	Miércoles	Jueves	Viernes
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					
17-18					
18-19					
19-20					
20-21					

Marque las asignaturas que desea seleccionar:

	Libres	Centro	Titulación	Curso	Cuatr.	Grupo	Creditos	Horario
Busquedas	<input type="text" value="Tod"/>	<input type="text" value="Tod"/>	<input type="text" value="Todos"/>	<input type="text" value="To"/>	<input type="text" value="To"/>	<input type="text" value="To"/>	<input type="text" value="Todo"/>	<input type="text"/>

Matriculación automática

La última opción disponible en la aplicación es la matriculación automática, para la cual todos los usuarios (profesores) habrán tenido que seleccionar previamente una selección de asignaturas favoritas, o predilectas entre las cuales le serán asignados sus créditos.

Marque sus asignaturas favoritas:

	Nombre	Centro	Titulación	Curso	Cuatr.	Grupo	Créditos
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	A	4.5
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	A	4.5
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	B	4.5
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	B	4.5
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	1	C	4.5
<input type="checkbox"/>	Introducción a la programación	FI	Ingeniería en Informática	1º	2	C	4.5

Este proceso debe ser iniciado por el administrador, y tiene la particularidad de que no siempre termina correctamente, entendiéndose por esto que a algún/algunos profesor/profesores se les deja sin matricular. Esto puede ser debido a que por el orden de jerarquías la aplicación no puede asignar ninguna asignatura al profesor en turno, siendo esto provocado, en la mayoría de casos, a que la lista de asignaturas favoritas no es suficientemente grande, o todos tienen una selección muy similar.

Instalación de la aplicación

Para el funcionamiento de aplicación necesitamos, antes de utilizarla, instalar un servidor, que será el que ejecute la lógica de la Web. Tendremos que instalar un servidor de aplicaciones Apache-Tomcat, en una versión 5.5 o superior. Este es de libre y gratuita distribución, obtenible siempre desde <http://tomcat.apache.org/>. Luego necesitaremos ‘desplegar’ el núcleo del servidor de la forma habitual, configurando todos los parámetros de url, etc.

También es necesario tener una base de datos MySQL, también de libre distribución. Debe encontrarse en el mismo ordenador que el servidor de la aplicación. Lo único que necesitamos en la base de datos es crear un nuevo usuario, con permiso para crear nuevas bases de datos, así como consultarlas y modificarlas. Si utilizamos el usuario `conexbd`, con contraseña `qwerty` no sería necesario configurar nada más, el servidor ya se encargaría de crear todas las tablas necesarias. Para la creación del usuario podemos utilizar la siguiente sentencia: `GRANT ALL ON *.* TO conexbd IDENTIFIED BY 'qwerty'`. Para linux es necesario especificar la conexión, por lo que la sentencia será: `GRANT ALL ON *.* TO conexbd@localhost IDENTIFIED BY 'qwerty'`;

En sistemas linux es necesario también crear en el directorio raíz una carpeta llamada `ficherosSubidos`, con permisos de lectura y escritura. Esta carpeta Realiza la función de almacén de los archivos subidos por el administrador. En Sistemas Windows no es necesario hacer esta carpeta, ya que el sistema operativo no restringe la creación de carpetas.

Una vez este instalado, Se puede acceder a la aplicación a través de una cuenta de administrador fija, con identificador ‘1’ y contraseña ‘1234’ con la cual se podrá comenzar la configuración de la aplicación.

Cualquier usuario sólo necesitará cualquier tipo de Terminal con un navegador Web estándar y conexión a la misma red. No obstante se recomienda una resolución de pantalla mínima de 1024 píxeles de anchura.

6. Ejemplos de uso

Para explicar mejor el funcionamiento de la aplicación realizamos un ejemplo de uso de ésta, donde podemos explicar mejor algunas consideraciones interesantes.

6.1. *Ejemplo con un administrador*

El administrador, tras entrar en la aplicación por primera vez, debe preparar la aplicación. Para ello accederá a la sección de 'gestión de información docente' y entrará en 'importar carga docente desde un archivo Excel'. Antes de cargar cada archivo debe tener las siguientes consideraciones:

- Los archivos están totalmente actualizados y no poseen información errónea o desfasada.
- En los archivos de personal se recomienda que solo aparezca el personal que va a formar parte de la asignación de docencia. Sino puede ser, deben ser eliminados manualmente de la aplicación.

Si la información de los Excel es correcta se procede a la carga de los mismos siguiendo un orden: 1º el archivo relativo a los profesores y el archivos de información docente. Si se va atender en cuenta los créditos por PDA entonces podemos cargar ese archivo, pero sino podemos pasar al siguiente paso, que es descargar un archivo Excel con toda la información obtenida, listo para que se le rellenen unos campos que antes no se han podido rellenar, que son los horarios, el edificio y el aula de cada puesto. Una vez este archivo se ha modificado se vuelve a subir en el último paso de carga de archivos, y con esto el sistema debería poseer toda la información docente necesaria.

Hay algunos datos que deberán ser modificados manualmente, como por ejemplo asignatura anuales asimétricas, con un cuatrimestre con más créditos que el otro. Nuestra aplicación reparte los créditos de forma simétrica en cada cuatrimestre, así que esto debe ser modificado manualmente.

El siguiente paso sería acceder a la configuración de correo si la asignación docente se va a desarrollar de manera deslocalizada, con cada participante en su ordenador, accediendo vía Web. Se recomienda probar la configuración introducida. Esto enviará un correo electrónico a la propia cuenta que se configuró. Si el proceso se desarrollará en una reunión, con una persona encargada de rellenar cada asignación docente, entonces este paso no es necesario.

Por último, llega el momento de configurar los parámetros del proceso, tales como la estimación de profesores para el año que viene, consideraciones con las asignaturas de Sistemas informáticos o de doctorado. Tras los cambios guarde la configuración y compruebe que las cuentas son correctas en las medias generadas. Esta media se calcula por ejemplo para los profesores:

total de docencia + PDA/*si se elige*/ - CorreccionCuentas

numeroProfesores + estimacionNuevasIncorporaciones

- epsilon

Si todas las cuentas cuadran, la aplicación ya está lista para comenzar el proceso. Si el proceso se desarrollará de manera deslocalizada se pulsa el botón Comenzar, y así comenzará el envío de emails a los participantes. Si la asignación la llevará a cabo un administrador solo deberá ir a la sección de elección de asignaturas y pulsar el botón para matricular desde la cuenta admin.

6.2. Ejemplo con un usuario

Un participante en el proceso de asignación docente, una vez empieza el proceso recibirá un mail con la información necesaria, como, por ejemplo, su identificador, contraseña y la dirección Web a la que acudir.

Con esta información el usuario ya puede acceder a la aplicación y ver las asignaturas que están disponibles, realizar búsquedas concretas y marcar las asignaturas que quiera. En un panel informativo, en la parte superior, podrá ver los datos relativos a la elección que lleva hecha, los posibles créditos de exención que le corresponden, los horarios para el curso siguiente, etc.

El usuario para poder matricularse deberá cumplir el mínimo exigido de créditos, y no incurrir en solapamientos de horario o en particiones de más de una asignatura anual, según exige el reglamento.

Si todo está bien, podrá matricularse y recibirá un correo que confirme su elección docente.

7. Bibliografía

Prácticamente todo el material de consulta que hemos necesitado se encuentra en Internet, habiendo sido Google la referencia para su búsqueda. Aquí se comentan las referencias más destacadas:

- Envío de ficheros a través de html con java:
<http://www.javazoom.net/jzservlets/uploadbean/uploadbean.html>
- Desconexiones seguras con jsp: <http://www.javaworld.com/javaworld/jw-09-2004/jw-0927-logout.html>
- Desarrollo de aplicaciones Web: <http://www.tic.udc.es/~fbellas/teaching/is/>
- WTP, tutoriales:
<http://www.eclipse.org/webtools/community/tutorials/SchoolSchedule/BuildingAScheduleWebApp.html>
- Programación con WTP:
http://www.programacion.com/java/articulo/jap_eclip_9/
- Plantillas jsp: <http://caucho.com/resin/doc/jsp-templates.xtp>
- Tutoriales de html, java, servlets y jsp:
<http://www.duffyinfodesign.com/support/>
- Java Servlets and JSP, Training and reference (libro gratuito):
http://www.murach.com/dloads/jsps/jsps_ch4.pdf
- Api de Javamail: <http://java.sun.com/products/javamail/>
- Empezar con JavaMail :
<http://www.chuidiang.com/java/herramientas/javamail/empezar-javamail.php>
- Tratamiento de ficheros Excel desde java:
<http://www.testpage.com.ar/2007/06/java-excel-jexcelapi.html>
<http://profesores.elo.utfsm.cl/~agv/elo330/2s05/projects/CeronSilva/presentacion.ppt>
- Diagramas UML:
<http://www.clikear.com/manuales/uml/>
<http://docs.kde.org/kde3/es/kdesdk/umbrello/uml-elements.html>
<http://elvex.ugr.es/decsai/java/pdf/3E-UML.pdf>

8. Palabras clave

- Casos de uso
- Docencia
- Jxl-Excel
- JavaMail
- JSP
- Log4J
- Login
- MySql
- Tomcat
- UML

9. Autorización para la universidad

Bajo el presente documento los abajo firmantes autorizamos a la Universidad Complutense de Madrid, y a cualquiera de sus representantes a distribuir y manipular libremente, con fines académicos, toda la información, así como cualquier tipo de desarrollo realizado bajo este proyecto de Sistemas Informáticos.

_____ Fdo: Beatriz Alonso Medina

_____ Fdo: Juan Luis Martínez Cesteros

_____ Fdo: Eduardo Renedo Gomero

